

**** *THE DAISY SYSTEM* ****

SOFTWARE ENSEMBLE

BY

BILL JONES

**** THE DAISY SYSTEM ****

SOFTWARE ENSEMBLE

BY

BILL JONES

The TS-2068 Software System

For The

LarKen Disk System

**Copyright 1989 by:
Gulf Micro Electronics
1317 Stratford Ave.
Panama City, Florida 32404**

** Introduction **

Daisy is a large software ensemble for the TS-2068 which contains several separate, but integrated, softwares. The Daisy system makes use of the DOS MERGE function extensively in order to control more programming than can be installed in TS-2068 memory at one time. All together Daisy controls about 80K bytes of programming, most of which remains in disk until called for.

Three Softwares of the Daisy ensemble are:
1. The Daisy Shell Manager. 2. The Smart Text DISK Data Base Manager and Word Processor. 3. The Mail Merge Software. Two more major softwares are Managed by Daisy but not supplied in the Daisy disk. These are Update Magazine's "D.Base-1" software and Bob Mitchell's "View Calc Disk".

In addition, the Daisy ensemble includes a group of Utility programs selectable from menu. These utility programs allow the user to print out the data files of other softwares such as Mscript, Tasword, Modem down load files, and generally any ASCII file that has been saved to disk.

This manual is intended to give you an overview of the Daisy Software, with enough detail to get you started creating and printing text data bases. The manual will only "touch the surface" of describing the total capabilities of Daisy. Your learning process will evolve as you use the software. Each new use will bring new discoveries for at least the next year. There is already quite a volume of documentation about Daisy given in articles in Update Magazine. Almost every issue of the magazine has at least one section of the Daisy software given as a separate operating program. We started in October 1987 with a complete listing and operational guide for the "Mail Merge Software", which is now a part of the Daisy ensemble.

Other articles in Update Magazine that give detailed operation and programming information about sections of the Daisy software are: "J-Utilities" Oct 87 issue, "List Looker" and "Purity" January 1988 issue, "MS-TAS" July 1988 issue, "Vars File SAVE LOAD" Jan 89. The program "TS-2068 Data Input" given in the April 89 issue of Update IS the Input and Edit functions of Daisy's main Text Data Base. Then, the July 89 issue of Update also has several of the Daisy program sections given. These are titled "D.Base-1", "LKDOS TOOLS", "Universal ASCII File Printer", and "View Calc for Disk". So, almost every issue of Update Magazine is an annex of this Daisy Manual.

More sections of the Daisy software will be published in future issues, not to sell the Daisy software, but to give good programming to the readers. So, you can associate Daisy with the daughter's wedding, ie, "something old, something borrowed, and something new, all packed together, and with a big frosted cake. And like a marriage, Daisy requires some patience and learning to make it work. But that isn't a departure from other large softwares. Some are so complex that there are 2 week schools on learning Lotus 1-2-3 or Word Perfect, both for the IBM PC. The last filer that I received quoted tuition for WP classes at \$450.00. So, let us approach the Daisy learning process with the understanding that it is in the

same class as those big softwares, and it won't be learned overnight.

But Daisy isn't as complicated to operate as those softwares mentioned, and I believe that with a bit of study and practice you'll be doing fairly good work with it in about a week's time. I worked hard to make Daisy a very "user friendly" software. But any programmer knows that the larger a program is the less friendly it becomes. It will help you to understand the concepts used in building this software, starting with the "Smart Text" annex of Daisy.

MENU DRIVEN SOFTWARE: Smart Text represents about seven years of on-going programming and updates. The Smart Text software "outgrew" the TS-2068's memory. So many functions were added that there was no FREE memory for DATA accumulation. Fortunately, Disk Drive systems were introduced for the TS-2068, which opened up new ways of programming. As with most softwares for the TS-2068, Smart Text originally used cryptic "command codes" which one had to either memorize or paste up on the wall for reminders of how to operate. Soon my computer room looked like a patch-work quilt with so many notes pasted everywhere. Now I ask you, if the programmer cannot remember all of them special key codes needed, then the user will likely have some problem, eh?

So, the first thing done with the DOS Versions of Smart Text was to change the concept to "MENU DRIVEN" software. All of the command codes were discarded and "subjective menus" used. Some 20 menus were created for Smart Text, and all SAVED TO DISK. These menus popped up from disk as needed for software operation. Daisy follows with about 30 menus in disk. The design concept is to "Present a subjective Menu when needed by the function that is in progress". Then each menu itself is informative. One does not need to refer to a crib sheet to find out that "Caps Shift while holding the Symbol shift and the Y.key" does a certain essential function.

As design work continued on the Daisy program, it too out-grew the TS-2068 Memory capacity, even after saving 18000 bytes of memory by using disk to store the menus. The next solution was to break the BIG program down into functional blocks and store the functions in disk. This concept involves leaving most of the software in Disk and MERGING sections in as needed.

So, while Daisy is already up to 80K in program length, it can still be made even larger, and the plan is to do just that. We will add more functions as the users ask for them, and keep on building until we run out of ingenuity. Daisy now controls three other softwares and manages the data products of almost all existing software for the TS-2068.

For now, Daisy provides a complete range of ADMINISTRATIVE functions. Its data bases are so very flexible that one can hardly find a needed format that cannot be created. When you think about it, 90 per-cent of the uses for a computer are "administrative", involving Data Base Management and Printing. Daisy is designed to encompass that 90%, to be loaded once in the morning to accomplish almost all of your daily Needs.

**** The Daisy System ****

The name DAISY was selected to be non-descriptive for a reason. This Daisy.B1 software is really a "System". Daisy encompasses Data Base Management, Word processing, and Disk Management, much in the manner of a DOS, or CP/M. The Daisy System contains more programming than the whole TS-2068 Computer, or for that matter, more than most whole computers provide in their built in ROM memory. Most of the programming is in Basic, but there are some Machine Code utilities used for providing SPEED where speed is needed. Basic programming provides easy "user access" to the running programs. One can almost always break into the programs by use of the BREAK Key to investigate the status of the computer or to use the computer for direct operation from the keyboard. This "open software" concept has both advantages and disadvantages.

One wouldn't think that such a large software as Daisy is would be able to accommodate break-in to allow the user to accomplish direct use of the computer. There are times when such freedom is very valuable. A good example would be when one is typing in data to a document and needs to insert a figure that can only be gotten by calculation. With Daisy this is simple to do. One just BREAKS and types <PRINT 4936/62>. The answer (79.6) is presented on screen. Then <GO TO fm ENTER> will resume Daisy operation. One can even MERGE a utility to Daisy, use it, and either let the utility remain attached, or Delete it. When finished <GO TO fm ENTER> resumes Daisy operation. No other software, especially large softwares, allow such freedom of user options.

But there is a down side of allowing user freedom of operations. Since Basic Report codes are not suppressed, every mistake in operation of Daisy will cause a REPORT CODE. This may be disconcerting at first during the learning process, but Report Codes are helpful. A report code allows you to analyze your mistakes and thus to learn how to operate more efficiently. Report Code stoppages would be terrible if they destroyed a portion of the data or if you couldn't recoup and resume operation. Daisy provides easy access back into the program, via the simple command "<GO TO fm ENTER>". There is no possibility of losing data by making a mistake that causes a report code. The only mistake that would cause the loss of data would be typing <CLEAR> or <RUN>. So, the first thing to learn about Daisy is <GO TO fm ENTER> to resume operation.

DAISY IS B-I-G!! It is the largest and most comprehensive software ever created for Sinclair Computers. Daisy DWARFS all other software for the TS-2068, is much bigger and more comprehensive than the QL's QUILL, and is a more comprehensive software than IBM's Word Perfect (that sells for several hundred bucks). The total program length of the Daisy system is about 80K bytes. But Daisy reaches outside of its own environment to use several other softwares for the TS-2068, such as Mscript, Tasword, D.Base-1, and View Calc. Daisy is a "Shell Manager" of about 8K in program length which manages 20 other programs in disk and 30 Menus of functions. Within all of these menus there are more than 200 choices of functions. As functions are elected from menu Daisy MERGES in the appropriate program. Then when the function

has been accomplished, Daisy deletes the function and presents another menu, thus keeping FREE operating memory at maximum.

The MERGE concept used by Daisy is assisted by a "Turbo" mode selectable at menu to allow certain often used functions to remain attached to the Daisy Shell program, thus eliminating the short pauses needed to merge in programs from disk. The use of TURBO ON mode reduces FREE Memory to around 21K. The only advantage of using TURBO is for rapid switching between Data Base management and printing functions. This type of operation is typical of writing and printing several different letters at one sitting. The TURBO OFF mode allows very large data bases to be created in memory at a small sacrifice in speed of switching between functions.

The Daisy Data Bases: Daisy is a Data Base Manager having such a variety of flexible data bases as no other software for any computer can match. There are SIX data bases and each has branches. These flexible data bases cover all conceivable needs for creating files for any purpose. Daisy provides complete DISK MANAGEMENT for each of the data bases. As files are created and edited they are stored in disk with discrete file names. Disk files may be re-loaded from disk for after editing. Disk files may be loaded for printing by the Daisy Word processor. The disk files may be printed singly or chained and printed in a continuous print out. With a few key punches you can schedule the print out of a thousand page book which will be printed non-stop. Or, a few key punches will schedule a letter to be printed to all names in one mailing list or several mailing lists, literally thousands of letters all perfectly addressed and ready to be stuffed into envelopes.

The MAIN TEXT DATA BASE: The selection "<1> Create - EDIT" enters this mode to create and edit the main text data bases. This is a dual data base having two names. We simply call them the "H" and "I" data bases, named after their data carriers. The data carriers are two Character arrays "H\$()" and "I\$()". These two arrays are dimensioned to the user's needs via prompts. You can think of these two data bases as "Paragraph Groups" that are each designed to store Edited Paragraphs, their length to be set by the user. When the first data group has been created and you are ready to tuck it away into memory storage a prompt comes to INPUT LENGTH. If you input, say, 1000, then the data base is set up for SEVEN groups of data, each 1000 characters in length. This will be the "H" data base. After the prompt, the data group that you have created is STORED as the "First of Seven".

As you proceed to create more data, six more STORE actions (in the "H" Data Base) are available. Then as more data is created, the next "I" Data base is set up and filled in similar fashion, first by prompt for Size, and then the Store Actions follow. After 14 STORE Actions have been accomplished the two data bases "H" and "I" are full. Typically this is about 14K of total data or about 6-8 pages of data. The data is in MEMORY and can easily be recalled to screen for editing again and again until it is perfect. Once both the "H" and "I" data bases are filled with 14 Stored groups of data, then Management decisions

are necessary. You can use the menus to PRINT the data, or you can use the menus to SAVE to disk. The SAVE to disk is with a discrete "numbered file". The discrete numbered file will later be used by the management system for controlled LOAD of data files for either editing or print-out.

THE MAIL MERGE DATA BASE: Here we have a separate software "Mail Merge" that is selectable at menu to create, edit, and manage Mail List Files. The selection for MERGING the Mail Merge software is at the OFFICE TOOLS Menu. You can create Mail list files of about 50 names and addresses. These files may be viewed, corrected, sorted, or SAVED for future use. Also, the Mailing list may be printed as a listing, or to labels, envelopes, or used immediately for addressing individual letters or chains of letters as selected by the Daisy menu. Upon exit of the Mail Merge functions the program itself is automatically Deleted from the Daisy shell.

THE OUTLINE DATA BASE: This is selectable at the Office Tools Menu for creating outlines of a type used in school schedules, church activities, or illustrative writing. The data base is completely flexible and can be set up to the user's needs. The theme is Major topics and Minor topics "under the Majors". Prompts allow you to set up as many Major Topics, and as many "Minor topics of each Major" as you desire. The outline prints out under page control with optional TAB settings.

"D.BASE-1" Data bases. Here we use plural because D.Base-1 has its own variety of data bases that can be created and managed. We are talking about a data base that contains "Records". Some of the various data bases are; Inventory records, Mail File records, numeric records, monthly appointment schedules by day, daily diary, etc. D.Base-1 files may be loaded to Daisy for including in its print products, mixed with text, outlines, etc. While D.Base-1 is not supplied with Daisy, the Daisy menus coordinates its load and operation.

SEQUENTIAL FILES Data Bases: This pair of Data Bases are for the LKDOS users, as no other TS-2068 system has such capability. Imagine! You may have a combination of other data bases in memory, enough to max out the FREE memory available. You can still create 50K more of data without disturbing the data that is in memory. You do it with Sequential files sent directly to disk as the files are being typed! AND! In two ways! One method used is to create short records and the other is to create Paragraphs of data. Lets take the "short record" method first. As you type a character counter is presented in the lower RH corner of the screen. You can decide the length of each record and type away using the counter as a guide. Each time ENTER is touched the record is sent to an opened disk file. You can just keep on truckin and build an enormous length file of records, or you can quit at any time. An optional print out is provided to print any sequential file that is in disk.

The second method of Creating a Sequential file is via use of the main Daisy INPUT section, which has FULL EDIT. Here we create paragraph length bursts of data to be sent to an opened Sequential file. Again our options are many. We

can create short files of text paragraphs, or we can create an enormous length file in disk. And again, Daisy provides an efficient print out of any disk sequential file. The option is to print BY LINE (for the short records), or "PRINT to ESTABLISHED FORMAT". In the latter case a few variables are set and the incoming paragraphs of data (from disk) are sent to the Daisy Word Processor for formatted printing.

The DAISY SYSTEMS MANAGER: Built into the Daisy Shell program is a very efficient Managing system that manages all Peripheral devices and their functions "in coordination with MENU Selections. The management system includes; The Display Monitor, The Disk Drive, and the Printer, to provide a MENU Driven System under the complete control of the user. The management systems revolves around the FUNCTION MENU, which is "Home Base". All of the 30 menus of Daisy evolve from and go back to The Function Menu. The Function menu provides a few direct actions, but it is mostly a Selection Menu for MAJOR Functions of the software. An extension of the Function Menu is gotten by selection of "<8> OFFICE TOOLS". These two menus provide access to all functions of the software, plus the coordination with three other softwares.

The concept is to provide the user with a "Menu for Choice" for whatever function that may be needed for use. Then sub-menus are provided as needed for follow-on selections. This is opposed to the "common way" that most softwares are driven; ie, requiring a page full of key combinations to be remembered or pasted on the wall. (Hit the Caps shift 4 key while holding Symbol shift to do something, etc, el. times 50 more!). The Daisy management system puts a menu before your eyes with subjective selections. AND, the menus are coordinated so that as major functions are elected, the next menu that appears will be related to the previously selected function.

PRINTER CONTROL and TEXT FORMATTING is achieved without "Junk embedded in text data", as most softwares require for commanding the printer and formatting a page. (/%&& to do a paragraph end and line space, etc.). Instead, Daisy gives you a FORMAT MENU where you establish the format desired. If a Print style needs to be established, the STYLE MENU lets you set your print style. By use of these two menus ANY style of print that your printer is capable can be set. During the menu selection process, automatic formatting is accomplished for "Center Printed text" with whatever print style and line length is elected. The theme is "automatically formatted pages of center printed text". However, offset printing can be selected at the Format menu, and Column Printing can be elected.

The DATA DISPLAY: The use of the Style and Format Menus (once) eliminates the need for inserting print control symbols within the textual data. Thus a data file is always a "pure ASCII File". During Input Typing and Editing the textual data is presented with all spaces visible in an un-formatted screen. There is no correlation at all with the appearance of the text on screen with the way it will be printed. Therefore you should NOT pad in spaces for formatting or line spacing. Instead just create

groups of data with proper sentence structure and let the automatic formatting features of Daisy do the printing. Daisy will format the paragraphs with a line space between. If you have elected to have indented first lines, then Daisy does that too. So, the screen display is used for only one purpose, and that is to show you the data so that you can edit it at will. A menu of editing functions is presented at the bottom of screen so that you don't need to remember any key codes.

DATA EDITING: INPUT TYPING: These two important functions are intertwined. Input typing uses a machine code program for very fast keyboard response. A " " underline cursor is presented ahead of the typing input. This cursor is movable with the arrow keys for editing. When an arrow key is used the fast typing input is escaped from and the EDIT mode is entered. At this time a set of menu options are given at screen bottom. There are so many editing functions that it is difficult to list them all. First, during INPUT typing, full EDIT is there. The menu at screen bottom appears anytime you shift out of INPUT into EDIT "via the use of an arrow key." You can place the cursor anywhere within the text and make deletions, type insertions, shift into BLOCK DELETE, or BLOCK INSERT. Passages of text that are deleted with BLOCK DELETE are held, and can be re-inserted with the BLOCK INSERT function. Or, you can type a long BLOCK INSERT.

During Input Typing, the last 148 characters typed are always present. As typing continues the screen size builds until you touch ENTER. ENTER shifts the typing up on screen making room for more. UP Arrow will pull down earlier generated data to the first line for viewing or editing. When EDIT action is finished (via touching ENTER) the INPUT typing mode is re-entered. An escape from INPUT is provided. The use of CAPS and SYMBOL SHIFT together shifts out of Input typing and brings the "QUICKIE MENU", which provides three more editing modes, plus other functional options. A study of the pictorials that follow will bring understanding of the variety of functions available.

The "L\$" CAPTURE: During EDITING, two functions use this CAPTURE CAPER to allow data to be "swapped about", deleted from one area and inserted at another place within the typing data—OR within any text group that has already been created and stored. A BLOCK DELETE results in TWO ACTIONS. The block of data is deleted, and the block of data is CAPTURED in the "capture buffer" which is "L\$". This data remains in L\$ until the NEXT BLOCK DELETE, upon which time the new data being deleted replaces the old. CAPTURE is a very powerful editing function that can be used in several ways. Many times you'll just want to DELETE a passage of text and discard it. Other times the passage of text may fit into another previously created data group very well. Why have to re-type a long insert?

After a data CAPTURE you can shift out of the INPUT mode, with CAPS SYM SHIFT, then elect "<2> RE-ED". Then the Quickie menu appears to allow you to select the previously stored data group for READ and EDIT. While Editing the cursor is moved to the place for "Inserting the captured data". Using " " to shift into BLOCK INSERT Mode, the choice <1> at screen bottom is "<1> INSERT L\$".

ZAPPO! It is done. Another common use of the CAPTURE (Block DELETE) is when you have exceeded the data length of the H and I data cells. Remember that YOU set the limits to these 14 data groups yourself. But even though you have set a length that suits your purpose, you will often type more data than can be STORED. Say for example, you have established 1000 characters as the pre-set length of the H Files.

The situation is, you have typed the third paragraph and upon reaching the Quickie menu for STORING the data to "H-3" the menu tells you that the data length is 1500 bytes. Well, you have a dilemma. If you elect to STORE, then the last 500 characters will be lost. So, what do you do to save that 500 characters for the next paragraph? The answer is to re-enter the INPUT mode and use the cursor TO BLOCK DELETE the last 500 characters. When you do, the 500 characters will be CAPTURED in the CAPTURE BUFFER (L\$). To do that you manipulate the cursor to skip up about 17 lines to find the beginning of a sentence to begin the 500 plus character Delete. Put the cursor there and exercise a BLOCK DELETE with CAPS+EDIT. There are two steps to a Block Delete. When you move to the Start of a group to Delete and Touch ENTER, a TOOT tells you that that spot has been MARKED. Next, you move the cursor to the END of the block of text to be deleted and touch ENTER again. ZAPPO! It is done!

BUT, we will do the Block Delete differently this time because we are DELETING TO THE END. Instead of having to move the cursor down 17 lines to the end of text, just touch ENTER. That shifts into TURBO Typing mode and the cursor is put to the right of the "Last Character" of the text. So, then all we have to do is use the LEFT ARROW key to move left ONE space. Then CAPS-EDIT ZAPS the 500 characters out. This DELETES the 500+ characters and puts them in the Capture buffer "L\$". Now you can return to the Quickie Menu and STORE the DATA, which has been reduced to under your limit of 1000. The 500+ characters are safe in the Capture Buffer.

We will continue the hypothetical problem. You now can STORE the 1000 character text group by using STORE at the QUICKIE MENU. AND, you'll want to begin your next paragraph with the data that is in the Capture Buffer. To do that we select "<1> INPUT Typing" at the Quickie Menu. This gets you the Input typing screen with just the cursor in the top left of screen. The way to enter the "BLOCK INSERT MODE" is to use an ARROW KEY to escape the Turbo Typing mode. BUT, in this case you have no data to move the cursor within. It is a BLANK screen. The only way to INSERT at the beginning of the SCREEN is to use the UP ARROW ONCE. You won't see any movement, but the bottom screen menu will appear. Then the shifted " " key will give you the INSERT MENU where a touch of <1> will insert the capture buffer at the beginning of the text screen. That was a long winded dissertation to tell you how to do a very useful "TRUNCATE the TYPING DATA" and INSERT the data in the next screen.

Important Notes on the USE OF VARIABLES

** The "varset.B6" Utility **

The VARSET.B6 program initializes the master Variable file for the Daisy program, when called for by selection of "<9> Re-initialize Daisy". This selection is at the Office Tools Menu. Then "varset.B6" boots in and re-sets the vars file and then deletes itself. The Daisy.B6 variable file is bytes in the variable file. The variables are carefully arranged so that the ones used in routines needing the greatest speed are initialized first in the TS-2068 variable file.

Within this system vars file there are four "pseudo hex" systems for the substitution of variables for commonly used numbers. This saves a tremendous amount of FREE memory within the program. Numbers ONE through 27 are represented by the "double OH" variables, ie, zero=oo, 1=oa, 2=ob, 3=oc, through oz=25. Then since the number 27 is used quite a lot, 27=zo. The next pseudo hex table is for "thousands", where k1=1000, k2=2000, through k9=9000. The third pseudo hex table is for "hundreds", where m1=100, m2=200, through m9=900.

The fourth pseudo hex table is for "tens", beginning with 30, where t3=30, through t9=90. (10 and 20 are already represented in the "oo" table by "oj" and "ou"). Sometimes during the programming these pseudo hex vars will be combined, as <GO TO k6+t9> which relates to <GO TO 6090>, or <GO SUB k2+m5> which relates to <GO SUB 2500>. A typical use of the "oo" vars would be <LET nn=nn+oa>, which increments the line counter nn by one (LET nn=nn+1). Other often used line numbers are represented by variables, as "fm=Line 2070". A perusal through the program lines, using <PRINT vars> can give you the whole table of Line numbers that are represented by variables.

So, the programming in the lines become quite cryptic, using variables almost entirely for numbers. Any examination of the program flow will require the relationship of variables to their number representations given above. There is good reason for the encoding of the numbers to vars. At one time I counted up over 8000 byts of FREE MEMORY that was saved within the Smart Text program, just by using variables for numbers within the programming. But all routines cannot be encrypted in such fashion. Since the use of a variable requires "look-up" time, the program operation is slowed. One example routine that must use "raw numbers" to avoid slow operation is the INPUT EDIT functions, where such use would cause an intolerable slow down.

There are also "transient" variables used throughout the program for calculations and as "switches". All of the single letter variables are temporary, and are used over and over by different routines. None of the single character vars have lasting effect, or affect from one routine to another. Incidentally, the "varset.B6" utility itself is over 2250 bytes in program length, which amount of FREE is conserved by the MERGE and DELETE concept.

Most of the MENUS have either a number or a Variable presented in the Right Bottom of the screen. These are the LINE NUMBERS of the Menu itself. EXAMPLE: the function menu has "fm" in the lower right corner. During a BREAK condition, <GO TO fm> will present the Function menu and the program will resume operation.

Some IMPORTANT variables are shown each time the Function Menu appears. These vars can be used to save "long trips" to other menus, with AN INTENTIONAL BREAK, and quick <LET>. These vars are: nn=Line Counter. In=pre-set page length. tb= TAB for left margin. ll=Line Length being formatted. qq=page center for centered print of captions. pg=current page number. To reset TAB from the existing displayed status to TAB 60: BREAK, type <LET tb=60: CONTINUE>. The Function Menu re-appears with the new TAB setting. Or, to change the page number to page 5: BREAK type <LET pg=5:CONTINUE>. The first case would allow OFFSET printing at TAB 60 (may need to change LINE LENGTH "ll" also). Such quick BREAKS and changes can speed up operation.

**** LETS GET STARTED! ****

1. LOAD the Software. You can do this by turning ON the Computer while holding the ENTER KEY. OR, type <RANDOMIZE USR 100:LOAD"Daisy.B6">. When the software loads there will be a series of PROMPTS to set up the software for your type of Interface and printer. Your printer should be ON for this. When the few prompts have been responded to the main FUNCTION MENU will appear on screen.

2. The FUNCTION MENU: While this Imposing menu is on screen would be a good time to look at pages 8 and 9 of the manual so that you can see the logic of the menu selections. Lets create some DATA, as almost all menu electives are for managing DATA. Touch <1> Data INPUT and the Quickie Menu will appear.

3. THE QUICKIE MENU: Touch <1> TYPING. A blank screen will appear with a Cursor in the top left. This is your invitation to begin typing data. Type some data, enough to fill about a half screen. As you type and make errors, delete the error with the Shifted Delete Key. But let some errors pass on by so that we can EDIT and correct them.

4. Now lets shift into the EDIT mode by using the UP Shifted UP ARROW KEY. Move the cursor about within the text with the four Arrow keys. Try DELETING a character. You do that by placing the cursor to the RIGHT of the character and using Shifted Delete. So far so good, eh? Notice that after deleting characters the ENTER springs the cursor back to the end of the text. This puts you back into the fast INPUT typing mode for more character generation.

5. Next, move the cursor up to the beginning of one of the sentences in the middle of the text. We will DELETE that Sentence. Place the cursor under the beginning character of the sentence and use caps shift and EDIT (the one key). A foot tells you that Daisy has MARKED that character as the beginning of a BLOCK DELETE. Now use the right arrow key to move the cursor UNDER the Last Character of the sentence. When there, again use the Caps Shifted One Key. The sentence should disappear from screen (a BLOCK DELETE).

6. A BLOCK INSERT: Now place the cursor up where the sentence was before it was Deleted, using the Arrow Keys. When there, use Symbol Shift to get the British Pound (on the CLEAR KEY bottom row). A new menu will appear at screen BOTTOM. Read the menu to digest its meaning, and then touch <1>. The previously DELETED sentence will now be inserted where it was to begin with.

7. A DIFFERENT BLOCK INSERT: This time, move the cursor to some place in the text, and get the Brit Pound character again with Symbol Shift and the pound key. Again the new menu will appear at screen bottom. This time touch <2>. The Quotes for string input will appear. Type some characters and then touch ENTER. This block of characters will be INSERTED from where the cursor was positioned. Now, for practice, BLOCK DELETE the last bunch of characters that you inserted.

8. ESCAPE FROM INPUT: Use Caps Shift and Symbol Shift together. The Quickie Menu will appear. Now that you have some data to work with,

the amount of characters will be reported in the Information section of the menu. Lets STORE THE DATA. Touch <3> STORE. A prompt will appear asking you to decide "The maximum Data Length for the paragraphs that you want to generate and STORE. A good figure to INPUT would be about 800. INPUT 800.

9. Now notice the new data presented at bottom left of the Quickie menu. "H=2" means that the NEXT data group that you STORE will go into the data base "H-2". Also notice that the "Cell Limit" is reported as 800, and the Typing Buffer is at Zero capacity. These information products guide you for subsequent data management.

10. Next, touch <1> to get back into the INPUT TYPING Mode. Again the blank Screen will appear. This time use the UP ARROW key ONCE. Then use Symbol Shift and the British Pound. The BLOCK INSERT Menu should come to bottom of screen. Touch <1>. The text data that you previously DELETED from the STORED Paragraph will be inserted. Type some more characters and then Move the cursor up to the to the Beginning of the next sentence. Again, use the Brit POUND. Touch <1> again. Again, the SAME data will be inserted, which illustrates that the "CAPTURE BUFFER" is preserved until "something else" is put into it.

11. Next, shift OUT of the INPUT Mode. Remember HOW? (Caps and Symbol Shift). This time when the Quickie Menu appears, touch <9> PASTE UP. Now type some characters and touch ENTER. Now you have inserted different character data into the CAPTURE BUFFER, and it will stay there until you make another BLOCK DELETE.

12. Touch <1> again. The previously typed data will appear on screen. Move the cursor up to the beginning of a sentence and get the Brit Pound again. Touch <1>, which will demonstrate the Block Insert to Paste-Up text.

13. Return to the Quickie Menu and STORE the paragraph. Then touch <2>. WOW! what a busy new menu! Study it a bit. This is the EDIT MENU for data that has previously been STORED. ENTER <1> to get the first paragraph that was Stored. Now run the cursor up to the beginning of a sentence. Get the Brit Pound for an Insert. Touch <1> and watch the data that is in the Capture buffer insert itself in the text.

14. Now touch ENTER, and watch the cursor spring to the END of the TEXT. Actually, it goes to END of 800 characters, and the last part is nothing but spaces. Use the UP arrow to position the cursor to two spaces beyond the end of the last sentence. Now use the Brit Pound for a Block Insert. Type a Insert and touch ENTER. This just demonstrates how you can recall previously STORED data, use the Block Insert to add on more text--of course "within the 800 character limit" that you input for data length. Now to get back to the Quickie Menu, use Cap and Symbol Shift.

15. Now to do something else. Since you have TWO data groups STORED, lets recall them and EDIT BOTH in sequence. At the Quickie Menu, touch <6>.

But the menu at bottom is different. This time you are in a "READ with EDIT CHOICE". Note that to EDIT, you touch the "E" key. But lets DONT. Just touch ENTER. Now you have the second group of text on screen. Touch "E" for Edit. Now you have the same type of Full EDIT functions previously found with the other menu. To get OUT of EDIT, use Caps and Symbol Shift. This brings the SAME text back to READ Again, and EDIT MORE if needed. Otherwise ENTER will get the third group of STORED data.

16. Now touch ENTER. Since you haven't yet STORED a third group of text, you will be viewing and EMPTY text cell. The capability is to View and EDIT 7 cells of data in each of the two data bases "H and I". Touch ENTER again and note that the information at bottom identifies the data as "H-4". A little trick to get back to the Menu is to Jab Break. Do that. Now you are back at the "read edit" menu where you could select <6> to view and edit the 7 groups of the "I data base". Instead, knowing that there is no data there, Enter to select the Quickie Menu.

17. Now back at the Quickie menu, select <1> Input again and type some characters. Use Caps Sym Shift to get back to the Quickie menu. This time touch <5>. You have just cleared the typing buffer.

18. Now to compete the sojourn through all Quickie Menu items, touch <7>. Now we will replace the word "The" with "Licorice" (just for a humorous kick). Enter the word "The" as the word to replace, and "replace with licorice". Have a cup of coffee or something. This is a slow routine, especially if you have all 14 data cells filled with long stored groups.

19. Item <0> is for EDITING a Sequential File that has been converted to a character Array. But we are not ready for that yet. Item "<:>" (colon key) lets you clear the data base and start fresh.

20. Now that we have some data to print, lets do it. Select <4> Help, which gets the Function Menu. Next select <8> Office Tools, and then <4> Print Menu. Here you will have choices of printing functions. Select <2> Letters. Follow the prompts and input 2 copies. This will be the acid test of whether you need to work on "Matching your Printer" to the software. If the letter prints out "center printed" with a "center printed letterhead" the HOORAY! If not then B-oweee. You have some work to do on your printer matching. There is a section of the manual about that subject.

You have become familiar with the management of "Just one" of the Daisy Data bases. This is a good start and you can use this for creating and printing Text Data files. The next step is to check out all of the functions of the FUNCTION MENU.

THE FUNCTION MENU

As you have seen, after a function has finished, the Function Menu RE-Appears. If you had trouble with the previous step, then a "report

code" would come at the screen bottom, which would indicate that you had either done something wrong, or your printer is not matched to the software. Any report code can be recovered from by typing <GOTO fm ENTER>.

Now at the Function Menu, lets take a Safari through the "Menu Wilderness". It will be educational. First touch <4> to see the Data Status Menu. This menu presents the status of all text data bases and the Mail File. Study in for a while. It is very informative, and is useful to help manage the data bases. Simple string data is also reported, but not managed by the software. To put data into simple string you need to BREAK and do a LET statement. When you have finished studying the Status Report just touch ENTER to return to the Function Menu. Now lets do that again with a different variation. Select <4> Status Report, and when the report comes JAB BREAK. This is the quickest way of "cleaning out all data" and starting afresh with a clean program.

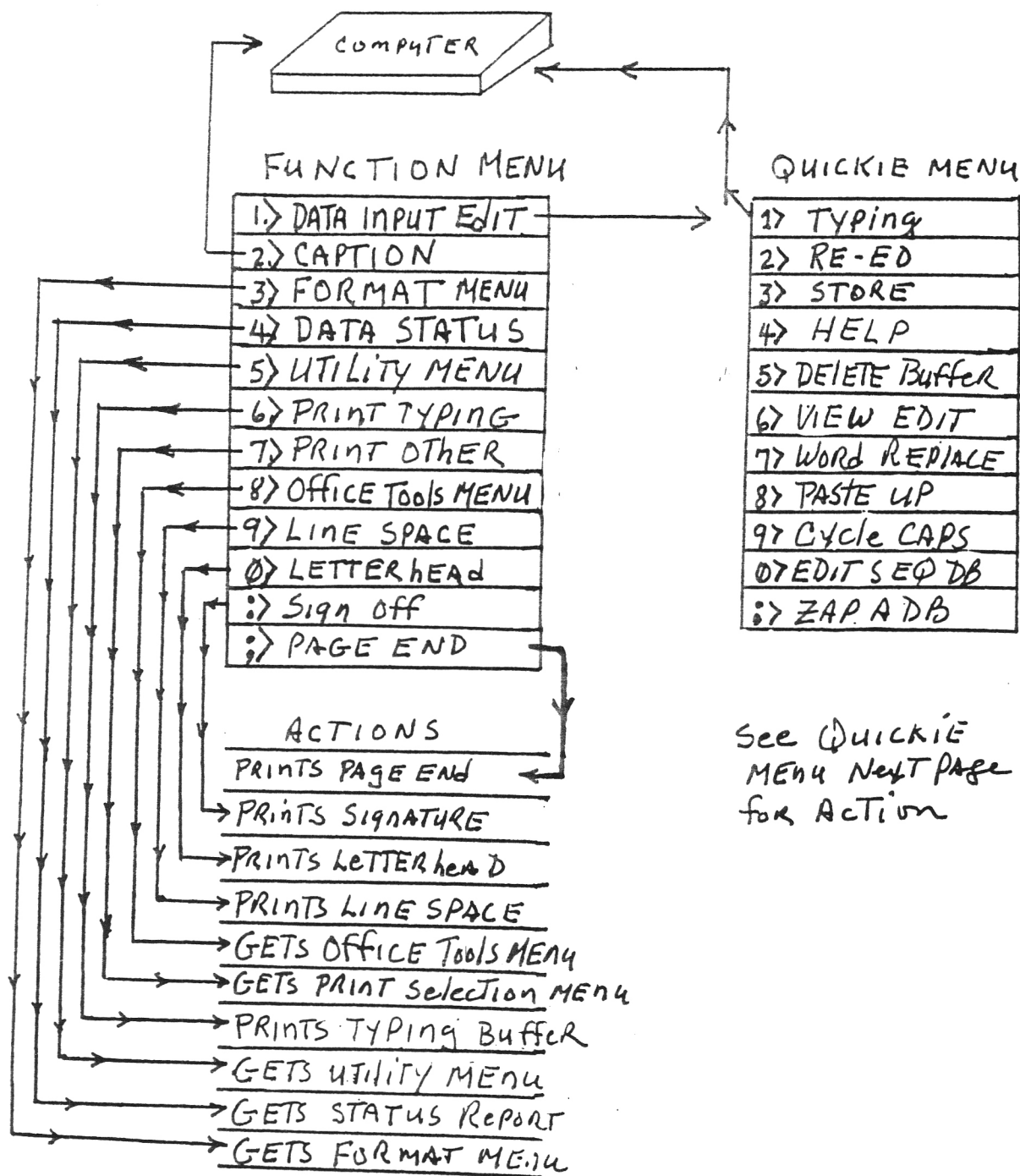
The selection <1> Yes will do that by doing a CLEAR, and then a re-load of the clean variable file "01697.C2". But lets dont do that right now because I want to use some of the data that is stored in your program. Instead ENTER <2> to return to the Function Menu. At the Function Menu, touch <5> for the Utility menu. The Utility Menu comes on with nine options. Study the menu for a while. Notice that item <5> offers the option of returning to the Function Menu. Most of the Daisy menus provide this option so that you can easily abort and get back to first base. Select <2> DELETE MENU. Now we have a menu that does many things. The first five selections deal with the data bases.

Each of the data bases may be CLEARED, or ALL may be CLEARED. The current status of FREE Memory is presented at top right. Item <5> is the same type of function that we discussed with the Status Report. A selection of <5> CLEARS all data and re-loads the clean vars file. Item <6> Turns "Turbo Off" and deletes all auxillary programs that may be "attached" to the main Daisy program. After that selection, the individual utilities MERGE to Daisy when called for by a Menu operation. This makes the software have short delays while the MERGE takes place, but provides a much larger FREE MEMORY capacity for large data bases.

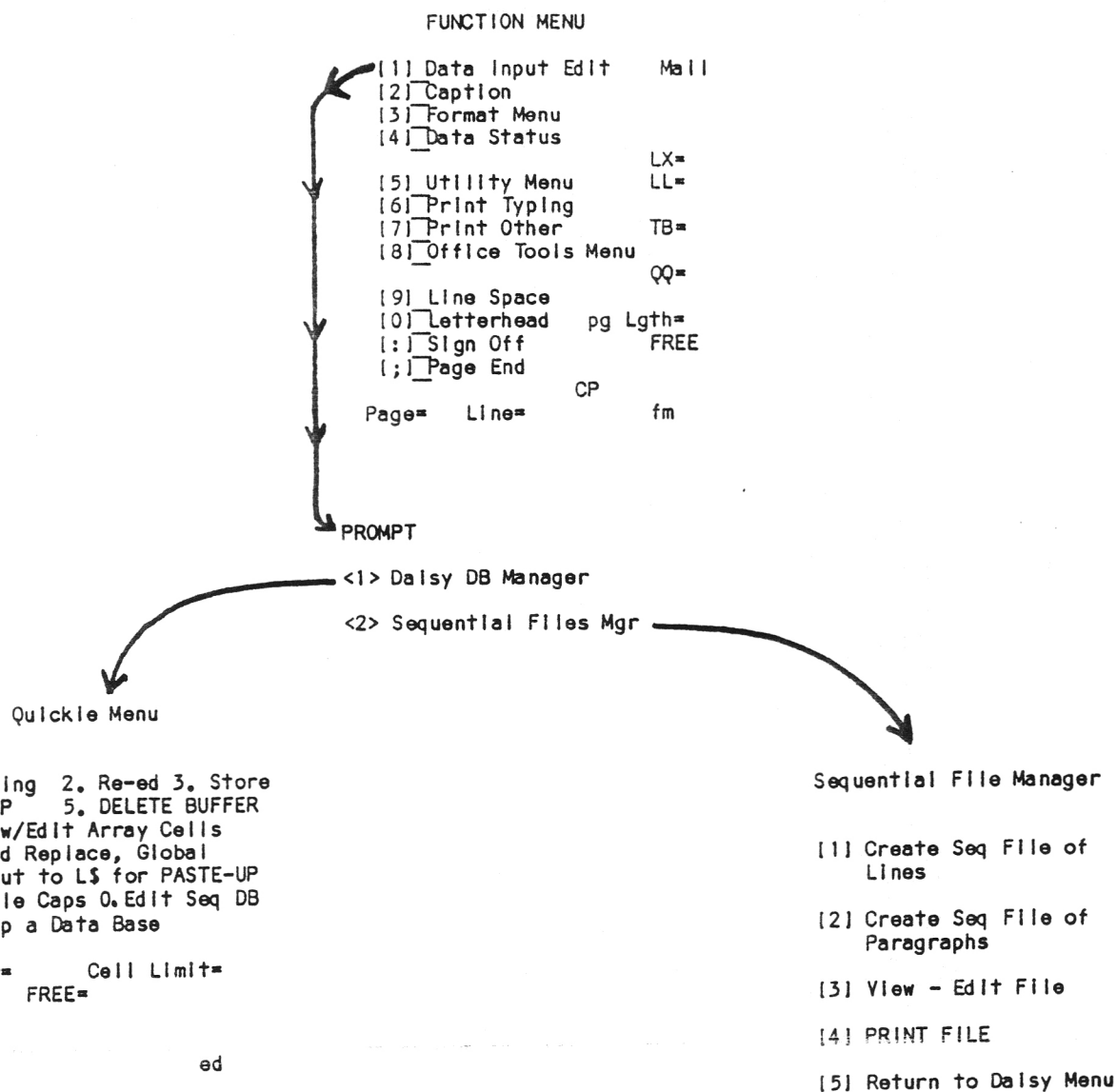
Now at the DELETE MENU select <8> to return to the Function Menu. At the function Menu, again select <5> Utility Menu. This time at the Utility Menu, elect <3> Load Menu. Here we have the ability to quickly LOAD in data files from disk. If you selected <1> H Array, the data loaded in would REPLACE the two data groups that you have in your program now. If you wanted to keep this data in program and LOAD a "I data base" then <2> would do it. Then you could print the two EXISTING data groups and the new group that may have 7 more data groups. The two groups would print in series (As 9 text groups).

Continue Investigating the SAVE Menu and others, working you way through all software menus. The Function Menu operates as a "rotary" to access all other menus, and is returned to as functions are completed. This cursory start should allow you to do simple data base input, Edit, and printing.

THE FUNCTION MENU



★ See Other MENUS
Succeeding PAGES



The Function Menu is "HOME BASE" for all of Daisy's Functions. You can always reach the Function menu from other Menus, OR, in a subscript BREAK condition, via <GO TO FM ENTER>.

The Illustration above shows the menu routing when "<1> Data Input-EDIT" is elected at the Function Menu. Daisy has several "Data Base Managers". The Quickie Menu (Left Illustration) is the main data base manager menu for the two Text Data Bases "H and I". The Sequential Files Data base is managed with the Menu at right.

The Illustrations that follow will deal with the coordination between the Function Menu and the QUICKIE MENU for Data Base Management.

*** Program "daz.B6" LLIST ***

7545 Bytes

The DAISY SHELL PROGRAM "Daisy.B6"

This Main LISTING is given mostly for reference by those who want to delve into the programming. It is not necessary for OPERATIONAL purposes to know the details of program line logic. But such detail is given as the "MERGE PROGRAMS" are discussed. This listing is the "DAISY SHELL MANAGER" program, which is 3 1/2 pages of condensed print. I will write in the side board margins to bring more information, mostly about CONCEPTS and operation.

AUTO
START

```

498 CLS : PRINT AT VAL "10",VAL "2";"Please Stand
By. . .": RETURN
500 RANDOMIZE USR 100: LOAD "dpr.C1"SCREEN$: PAU
SE 0: LET z=CODE INKEY$-48
502 IF z<1 OR z>3 THEN GO TO VAL "502"
504 IF z=1 THEN RANDOMIZE USR 100: MERGE "varsv. — 28
B6": LET alt=508: GO TO 9902
506 IF z=3 THEN GO TO fm
508 RANDOMIZE USR 100: LOAD "dprtr.B6" — 29
600 RANDOMIZE USR m1: MERGE "cat.B6": GO TO VAL " — 37
9822"
700 RANDOMIZE USR 100: MERGE "scopy.B6": GO TO VA — 30
L "7004"
840 GO SUB VAL "495": RANDOMIZE USR m1: MERGE "tb — 25
.B6"
900 ON ERR RESET : ON ERR GO TO VAL "902": DELE
TE VAL "842",VAL "877": STOP
902 ON ERR RESET : GO TO alt
1000 RANDOMIZE USR m1: LOAD "scopy.B6" — 30
1895 CLS : GO SUB VAL "495": RANDOMIZE USR m1: MER — 32
GE "mail.B6": GO TO k2
2057 DELETE VAL "1896",VAL "2055": GO TO fm
2060 INK 7: LET turbo=oa: POKE VAL "23730",VAL "87 — 14
": POKE VAL "23731",VAL "255": RANDOMIZE USR m1: M
ERGE "init.B6"
2069 DELETE VAL "2061",VAL "2068"
2070 ON ERR RESET : RANDOMIZE USR m1: LOAD "fm.C1
"SCREEN$: LET seq=oo: LET alt=fm: LET gg=oo: LET
nt=nz: LET mzk=oo
2071 PRINT AT op,VAL "28": INK oe;ln;AT og,zo;("ON
" AND turbo>oo)+("OFF" AND turbo<oa);AT or,oz; FR
EE ;AT ot,ot;cp;AT ob,VAL "26";("pica" AND ps=oa)+
("Elite" AND ps=oc)+("Cond" AND ps=od)+("PICA" AND
ps=ob);AT oh,VAL "27";lx;" ";AT ol,VAL "28";ll;AT
ok,VAL "28";tb;AT om,VAL "28";qq;AT ot,og;pg;AT o
t,op;nn;AT oe,VAL "26": INVERSE oa;("Yes" AND LEN
o$(oa,oa)>oa)+("No " AND LEN o$(oa,oa)<ob); INVER
S E oo: GO SUB sq: PRINT AT oo,VAL "27": INK oe; FLA
SH oa;"a"; INK og: GO SUB ll: IF z<oo OR z>ok THEN
GO TO VAL "2071"
2072 IF z=ok THEN GO SUB pe: GO TO fm
2073 IF z=ol THEN BEEP ob/oj,ov: GO SUB sp: GO TO
VAL "2071"
2074 IF z=of THEN LET w$=u$: GO SUB te: GO TO fm
2075 GO TO (z=oa)*VAL "3416"+(z=ob)*VAL "2094"+(z=
oc)*VAL "2078"+(z=od)*VAL "3579"+(z=oe)*VAL "4099"
+(z=oh)*VAL "4050"+(z=og)*VAL "3700"+(z=oo)*VAL "2
592"+(z=oj)*VAL "4090"
2076 LPRINT !: LET nn=nn+oa: IF nn>=ln THEN GO SU
B pe
2077 RETURN
2078 GO SUB VAL "498": RANDOMIZE USR m1: MERGE "fo — 22
mat.B6"
2092 DELETE VAL "2080",VAL "2091": GO TO fm
2093 DELETE VAL "2080",VAL "2091": GO TO k8
2094 CLS : PRINT AT oj,od;"Type the Cap" TAB od;"
Ln lgth limit ls: ";lx: INPUT m$: CLS
2095 LPRINT TAB qq-LEN m$/ob;m$: LET nn=nn+oa
2096 IF gg=oa THEN RETURN
2097 GO TO fm
2098 LET tel=VAL "3742": IF turbo>ob THEN CLS :
GO SUB VAL "498": RANDOMIZE USR m1: MERGE "repp.B6 — 19
"
2180 CLS : LET rpr=oa: PRINT AT oo,oj;"Event # ";l
: LET zp=zx: IF turbo>ob THEN GO SUB VAL "498": — 20
RANDOMIZE USR m1: MERGE "usrpgm.B6": RANDOMIZE USR
m1: MERGE "wdpro.B6" — 21
2181 CLS : PRINT AT oj,oo;"Copy # ";l: REM LET m$
="HEADER": GO SUB VAL "2095": GO SUB sp
2410 ON ERR GO TO VAL "2411": IF turbo>ob THEN
DELETE VAL "2100",VAL "2174": DELETE VAL "2182",VA
L "2403": DELETE VAL "3742",VAL "3812": STOP

```

MENUS AND TEXT
ARE NOT Related.

MENUS SUPPORT LISTING

AUTO START Line is 2060

SCREENS Load
From Disk

FUNCTION MENU	
11 Data Input Edit	Mail
12 Caption	
13 Format Menu	
14 Data Status	
15 Utility Menu	LX=
16 Print Typing	LL=
17 Print Other	TB=
18 Office Tools Menu	QQ=
19 Line Space	
10 Letterhead	pg Lgth=
1 Sign Off	FREE
1 Page End	
Page=	Line= CP fm

<GO TO FM ENTER>

"The CURE To What Ails You
BETT'RN PePTABismo
FOR Subscript ERRORS.

```

2411 ON ERR RESET : ON ERR GO TO VAL "2412": DEL
ETE VAL "4055",VAL "4088": STOP
2412 ON ERR RESET : GO TO fm
2550 PAUSE oo: LET t$=INKEY$: IF t$="" THEN GO TO
11
2552 LET z=VAL "-48"+CODE t$: RETURN
2554 PAUSE oo: LET t$=INKEY$: IF t$="" THEN GO TO
1k
2556 IF t$<>"y" AND t$<>"Y" AND t$<>"n" AND t$<>"N
" THEN GO TO 1k
2558 IF t$="y" OR t$="Y" THEN LET z=oa
2559 IF t$="n" OR t$="N" THEN LET z=ob
2560 RETURN
2570 CLS : LET m$="SM SAVE": LET w$="SM": GO SUB
k5+m2: INPUT "Input File Nbr: ";c: LET w$="SM"+STR
$ c+"B1": RANDOMIZE USR m1: SAVE w$ LINE VAL "206
0": RANDOMIZE USR m1: SAVE "cat,C1"CODE VAL "24311
",VAL "265": GO TO VAL "5120"
2574 BEEP oa/ob,od: BEEP oa/ob,oh: BEEP oa/ob,oj:
BEEP oa/ob,od: GO SUB sq: RETURN
2576 INK xa: PLOT oo,oo: DRAW oo,VAL "175": DRAW V
AL "255",oo: DRAW oo,-VAL "175": DRAW -VAL "255",o
o: LET xa=xa+oa: IF xa>og THEN LET xa=oa
2577 INK xa: PLOT oh,oh: DRAW oo,VAL "159": DRAW V
AL "239",oo: DRAW oo,-VAL "159": DRAW -VAL "239",o
o: LET xa=xa+oa: IF xa>og THEN LET xa=oa
2580 INK og: BEEP oa/oj,VAL "40": RETURN
2590: RANDOMIZE USR m1: MERGE "lh.B6": RETURN
2592 RANDOMIZE USR m1: MERGE "lh.B6": GO TO VAL "4
055"
2594 DELETE VAL "4055",VAL "4088": GO TO fm
2598 RANDOMIZE USR 100: LOAD "vcalc.B6"
2600 CLS : IF turbo<>oa THEN LET turbo=oa: LET al
t=VAL "9668": GO SUB VAL "395": RANDOMIZE USR m1:
MERGE "tb.B6": GO TO VAL "850"
2603 GO TO VAL "9668"
2604 IF turbo<>oa THEN DELETE VAL "9668",VAL "973
7": ON ERR GO TO VAL "2605": DELETE VAL "3742",VA
"3812": STOP
2605 ON ERR RESET : GO TO alt
3679 RANDOMIZE USR m1: MERGE "st.B6": GO TO VAL "3
580"
3683 ON ERR RESET : DELETE VAL "3580",VAL "3584":
GO TO k6+t5
3689 DELETE VAL "3580",VAL "3584": GO TO fm
3700 RANDOMIZE USR 100: MERGE "pt.B6"
3719 DELETE VAL "3701",VAL "3718"
3741 IF turbo<>ob THEN RANDOMIZE USR m1: MERGE "w
dpro.B6"
3814 IF turbo<>ob THEN DELETE VAL "3742",VAL "381
0"
3816 RETURN
3900 LET alt=VAL "3902": GO SUB VAL "495": LET tur
bo=oo: RANDOMIZE USR m1: MERGE "tb.B6": GO TO VAL
"850"
3902 CLEAR
3904 RANDOMIZE USR VAL "100": MERGE "btls.B6": GO
TO VAL "3912"
4041 DELETE VAL "3912",VAL "4040": RANDOMIZE USR V
AL "100": MERGE "cho.B6": GO TO VAL "3912"
4042 DELETE VAL "3912",VAL "4040": RANDOMIZE USR V
AL "100": MERGE "mzrt.B6": GO TO VAL "3912"
4043 DELETE VAL "3912",VAL "4040": GO TO VAL "3904
"
4044 ON ERR RESET : DELETE VAL "3912",VAL "4040":
RANDOMIZE USR VAL "100": MERGE "varset.B6": GO TO
VAL "9994"
4046 DELETE VAL "3912",VAL "4040": GO TO VAL "3904
"
4050 ON ERR RESET : RANDOMIZE USR m1: LOAD "ot.C1
"SCREEN$: LET zy=oo: LET zx=oo: LET zm=oo: LET mz
=oo: LET az=oo: LET no=oo: LET nx=oo: LET z5=oo: L
ET df=oo: LET scp=oo: LET mm=oo: PRINT AT od,ow; I
NVERSE oa;"turbo=";turbo; INVERSE oo

```

** General Information **

The DAISY SHELL is just that. It contains the "Software Manager" and some utilities that stay aboard. FREE MEMORY upon loading is about 21K. The general concept is to break the large software into about 25 functional elements, all stored in disk. As the elements are needed for operation they MERGE IN to the main SHELL program. When the functions that are inclosed in the MERGE elements operate, the trip to the Function Menu DELETES the "Merge.B6" program, thus keeping FREE MEMORY at Maximum.

The TURBO MODE: Early on it was discovered that programs that are longer than about 2K in program length take about 12 seconds to complete a MERGE-IN. 12 seconds does not sound like a long time, but when doing data management it seems like an eternity. So, a TURBO Mode was devised, where the Data Base INPUT and EDIT functions are "stuck to" the Shell program. Also, since a close cousin to the Data base functions is PRINTING, the printing functions are "attached" when in the TURBO 2 mode.

When first LOADED, via <LKDOS AUTO LOAD> or via <RANDOMIZE USR 100:LOAD "Daisy.B6"> the mode is TURBO 1. We actually use a "variable switch" to denote TURBO STATUS. The switch is variable "turbo". "turbo=0" when in the OFF mode and "turbo=1" for DATA BASE Management, Turbo 2 for PRINTING. FREE MEMORY is reduced to about 21K. But, you don't have to wait for the most often used functions to MERGE in during switching back and forth between the Function Menu, the INPUT section, the EDIT section, and the PRINTING sections.

You, the operator, have control of the TURBO MODE from two Menus. To turn Turbo OFF, visit the OFFICE Tools Menu. The route is from the Function Menu item <8>, Then ITEM <1> at OFFICE Tools. The result of selecting TURBO OFF is that these

DATA STATUS

```

AS=0      par In H$=
BS=0      par In I$=
CS=0
DS=0      FREE=
ES=0
FS=0      Ent Whn Rdy
GS=0

```

```

LS=0
QS=0
RS=0
JS=0
SS=0
US=0

```

NO
MAIL
DATA

Use K\$, P\$, V\$ for spare
Data ARRAYS 3579

See office
Tools Menu
Next Page

```

4051 GO SUB sq: GO SUB ll: IF z<ov AND z>oi THEN
LET z=z-og
4052 GO TO (z>ok)*VAL "4051"+(z=oe)*VAL "840"+(z=ob)*VAL "500"+(z=oc)*VAL "6700"+(z=od)*VAL "2098"+(z=oe)*VAL "1895"+(z=of)*VAL "2598"+(z=og)*fm+(z=oh)*VAL "5400"+(z=oi)*VAL "9994"+(z=oj)*VAL "2600"+(z=ok)*VAL "3900"+(z=oo)*VAL "8120"
4090 LET qq=(qq AND cp=oo)+(VAL "34" AND cp=oa)+(VAL "98" AND cp>oa): LPRINT : LPRINT : LPRINT : LPRINT TAB qq;n$(ok): LPRINT : LPRINT : LPRINT : LPRINT TAB qq;n$(om): LET nn=nn+oh: IF scp=oa THEN GO SUB k8+m2
4091 IF rpr=oo THEN GO SUB pe
4092 IF gg=oa THEN RETURN
4094 IF rpr=oa THEN RETURN
4098 GO TO fm
4099 RANDOMIZE USR 100: MERGE "dc.B6": GO TO VAL "5160"
5212 DELETE VAL "5100",VAL "5210": GO TO (z<oe)*VAL "3579"+(z=oe)*fm+(z=of)*VAL "600"+(z=og)*k1+(z>og)*VAL "3900"
5213 DELETE VAL "5100",VAL "5210": GO TO (z=oe)*VAL "9900"+(z=of)*fm
5214 DELETE VAL "5100",VAL "5210": GO TO (z=oe)*VAL "6050"+(z=of)*VAL "840"+(z=og)*VAL "3579"+(z=oh)*fm
5215 DELETE VAL "5100",VAL "5210": RANDOMIZE USR m1: CAT ".A$",: PRINT #oo;"At STOP, ENTER Type LOAD Command": PAUSE k1: STOP
5216 DELETE VAL "5100",VAL "5210": GO TO alt
5217 DELETE VAL "5100",VAL "515210": RANDOMIZE USR m1:: GO TO VAL "9900"
5218 DELETE VAL "5100",VAL "5210": GO TO fm
5300 CLEAR VAL "65367": DELETE VAL "5226",VAL "5290": RANDOMIZE USR VAL "100": POKE VAL "16090",VAL "200": RANDOMIZE USR VAL "100": POKE VAL "16094",VAL "0": LPRINT : RANDOMIZE USR VAL "100": LOAD "cc1r.C1"CODE : DIM k$(8): DIM v$(VAL "1630"): LET k$="01638.C2": RANDOMIZE USR VAL "100": MERGE "varsy.B6": GO TO VAL "9990"
5400 RANDOMIZE USR m1: MERGE "pg.B6"
5410 DELETE VAL "5401",VAL "5408": GO TO fm
5890 RANDOMIZE USR m1: MERGE "pe.B6"
5914 DELETE VAL "5900",VAL "5913": RETURN
6050 ON ERR RESET : CLS : PRINT AT VAL "10",VAL "0";"Complete DATA DELETE and LOAD of Variable File was Ordered? CONFIRM" <1> Yes -or- <2> No": PAUSE NOT PI: LET z=CODE INKEY$-VAL "48": IF z<oa OR z>ob THEN GO TO VAL "6050"
6052 IF z=oa THEN CLEAR : RANDOMIZE USR VAL "100": MERGE "varsy.B6": POKE VAL "24374",VAL "22": POKE VAL "24375",VAL "8": LET k$="01697.C2": DIM v$(VAL "1685"): GO TO VAL "9982"
6054 GO TO fm
6700 CLS : RANDOMIZE USR m1: LOAD "tbo.C1"SCREEN$
6701 GO SUB sq: GO SUB ll: IF z<oa OR z>oc THEN GO TO VAL "6701"
6702 IF z=oc THEN GO TO fm
6704 IF z=ob THEN GO TO VAL "6712"
6705 CLS : PRINT AT oj,ob;"Please Identify the number of TAB ob;"This PROGRAM DISK": INPUT "Input Disk Number":d: POKE VAL "24373",d
6707 RANDOMIZE USR m1: MERGE "varsy.B6"
6709 POKE VAL "24375",VAL "26": POKE VAL "24374",VAL "51": GO TO VAL "9902"
6710 POKE VAL "24375",VAL "26": POKE VAL "24374",VAL "51": PRINT AT VAL "18",NOT PI: FLASH NOT NOT PI;z$: FLASH NOT PI;" Is the file name SAVED"Make Note, then ENTER": LET d=PEEK VAL "24373": RANDOMIZE USR VAL "100": GO TO d
6711 CLS : PRINT AT VAL "18",NOT PI: FLASH NOT NOT PI;z$: FLASH NOT PI;" Is the file name SAVED"Make Note, then ENTER": LET d=PEEK VAL "24373": RANDOMIZE USR VAL "100": GO TO d

```

"Merge.B6" programs are DELETED. Then each function that requires one of the ".B6" programs will result in a 3-12 second wait for the merge to be accomplished.

Select Data to Print

<1> H\$(1)	<8> I\$(1)
<2> H\$(2)	<9> I\$(2)
<3> H\$(3)	<10> I\$(3)
<4> H\$(4)	<11> I\$(4)
<5> H\$(5)	<12> I\$(5)
<6> H\$(6)	<13> I\$(6)
<7> H\$(7)	<14> I\$(7)

15> Typing	18> C\$
16> A\$	19> D\$
17> B\$	20> E\$

21> Abort

Enter Choice: prt

OFFICE TOOLS

[1] Turbo Change turbo=1
[2] Print Code ASCII File
[3] Load Dbase=1
[4] Repeat Print Menu

[5] Load Mail Merge
[6] Load View Calc
[7] To Function Menu
[8] Page Management

[9] Re-initialize Daisy
[10] Create/Print Outline DB
C [A] LKDOS Seq DB Manager
A [B] LKDOS Desk Top
P (Temp Substitute)
S

4050

TURBO ON is selected by item <1> at the Office Tools Menu. When selected, the programs, "Inpt.B6" merges. This is the fast INPUT typing program and the EDIT functions.

"Turbo 2" mode was added for printing speed. When in Turbo 2, the "REPP.B6", "USRPM.B6" and "WDPRO.B6" programs are attached. And the data base programs are deleted.

The advantages of using TURBO 1 is its speed of Data base Management and Printing selections. Each function does not operate faster, but the transfer between functions are streamlined. The practical data capacity for TURBO ON is about 2/K.

The advantages of operating in TURBO OFF mode is a much greater Data capacity, about 30K, and the ability to operate ALL of the ".B6" annexes without fear of compromising FREE MEMORY. You can have quite a large


```

6712>RANDOMIZE USR VAL "100":LOAD "dbase1.B6"
7092 DELETE VAL "7004",VAL "7090": GO TO fm
8000 CLS : RANDOMIZE USR m1: MERGE "stymn.B6" — 23
8114 ON ERR GO TO VAL "8115": DELETE VAL "8001",V
AL "8110": DELETE VAL "9202",VAL "9208": DELETE VA
L "9500",VAL "9520": STOP
8115 ON ERR RESET : GO TO VAL "2078"
8120 INPUT "Install Pgm Disk and ENTER #";dd: RAND
OMIZE USR m1: GO TO dd: RANDOMIZE USR m1: MERGE "o
tln.B6" — 26
8176 DELETE VAL "8121",VAL "8168": GO TO fm
8200 LPRINT '': LET nn=nn+1: LET w$=u$: GO SUB +
e: RETURN
8990 RANDOMIZE USR m1: MERGE "df.B6" — 27
9024 DELETE k9,VAL "9022": GO TO VAL "2100"
9025 DELETE k9,VAL "9022": GO TO VAL "4050"
9380 RANDOMIZE USR m1: MERGE "word.B6" — 27
9492 DELETE VAL "9400",VAL "9488": GO TO (turbo>oo
)*ed+(turbo<oa)*VAL "3416"
9866 DELETE VAL "9822",VAL "9864": GO TO fm
9900 RANDOMIZE USR m1: MERGE "varsv.B6" — 28
9992 DELETE VAL "9902",VAL "9991": GO TO PEEK VAL
"24374"+VAL "256"*PEEK VAL "24375"
9994 POKE VAL "24375",VAL "8": POKE VAL "24374",VA
L "22": RANDOMIZE USR VAL "100": MERGE "varset.B6" — 31
GO TO VAL "9992"
9995 POKE VAL "24375",VAL "8": POKE VAL "24374",VA
L "22": RANDOMIZE USR VAL "100": MERGE "varsv.B6": — 28
GO TO VAL "9992"
9996 POKE VAL "24375",VAL "8": POKE VAL "24374",VA
L "22": RANDOMIZE USR VAL "100": MERGE "varsv.B6": — 28
GO TO VAL "9992"
9997 GO TO VAL "3010"

```

END of DAISY SHELL MANAGER

Some Valuable Hidden Functions

Normally the ".B6" annex programs are NOT ON BOARD the Daisy Shell program. The .B6 programs are automatically MERGED AS NEEDED. Sometimes it may be useful to BREAK IN "after" a merge has taken place to make special one-time edits of the merged .B6" program. But such editing actions will have only temporary effect. When the .B6 program "leaves the shell", the next MERGE of the same program will not have your modifications. To make permanent changes, the .B6 program must be loaded and changed, then SAVED OVER the original .B6 program.

*** SOME KEY <GO TO's>: For use when the program has been Interrupted: <GO TO FM> gets the Function Menu. <GO TO ed> gets the EDIT Menu. <GO TO 4050> gets the Office Tools Menu. <GO TO 2060> Re-Initializes the Daisy program without data loss. <GO TO 6050> Re-loads the clean vars file.

NICE TO KNOW, and sometimes useful: <GO TO pe> executes a page end. <LET W\$="data":GO TO te>- If TURBO is OFF this will MERGE the word processor and print W\$ in the current format. <LET W\$="data":GO SUB tel, if turbo is ON will print W\$ data. <LET W\$="data":GO TO 2096> will print a centered header. <LET m\$="data":GO SUB 2093>, if in a program line in the User Programming Group, will print a centered caption ahead of the data to be printed. Line number 2181 is a convenient place to schedule the "leading header" for the whole text group.

data base in memory and still operate all of the ".B6" annex functions when TURBO is OFF. So, you should always be cognizant of the TURBO MODE that you are operating with, and TURN TURBO OFF when not involved in data base input, edit, and printing. It is very easy to reverse the TURBO Mode at menu.

LOAD MENU

1. Load In H\$ Array file
2. Load In I\$ Array file
3. Load In Mail List
4. Load In a Var file
5. Other type of LOAD
6. To Function Menu

5100

Current Page is pg 1

Line to pg end=52

Select

- <1> Page end
- <2> Adjust line to pg end
- <3> Re-num Page
- <4> To fun menu

pm

Utility Menu

- 111 SAVE Menu
- 121 Delete Menu
- 131 LOAD Menu
- 141 Status Report
- 151 Function Menu
- 161 Copy Catalog
- 171 Screen Copy
- 181 Number Tables
- 191 Music Interlude

5160

DELETE MENU

- 111 Mail Data
- 121 H\$ Data Base
- 131 I\$ Data Base
- 141 Spare Stores
- 151 All Data
- 161 All Merge .B6 Programs
- 171 To Status Report
- 181 To Function Menu 5168

SAVE MENU

- <1> DAISY Prog and Data
- <2> Mail Data
- <3> H\$ Data
- <4> I\$ Data
- <5> Disk Save All Data
- <6> To Function Menu

5164

"Init,B6" LKDOS Initialization

```

2061 BORDER oo: PAPER oo: INK og
: CLS : RANDOMIZE USR m1: LOAD "
cpr.C1"SCREEN$: RANDOMIZE USR m
1: LOAD "cat.C1"CODE : PRINT #oo
;"PRESS A KEY...": GO SUB VAL "2
574": PAUSE oo: CLS : PRINT AT o
j,od;"Touch:"TAB od;"<1> For T
S-2040 Printer"TAB od;"<2> For
Dot Matrix"TAB od;"<3> For Da
lisy Wheel": GO SUB sq: GO SUB ll
: LET pr=z-oa: IF z<oa OR z>oc T
HEN GO TO VAL "2061"
2062 IF pr=oo THEN RANDOMIZE US
R m1: OPEN #og,"LP": LET ll=VAL
"32": LET qq=VAL "16": LET lx=ll
: LET tb=oo: RANDOMIZE USR m1: P
OKE VAL "16092",oo: GO TO VAL "2
069"
2063 RANDOMIZE USR m1: LOAD "lf.
C1"SCREEN$: GO SUB ll: GO SUB s
q: LET a=z-oa: PRINT #oo;"Printe
r Need Line Feed? <y> <n>": GO S
UB lk: GO SUB sq: LET b=z-oa: RA
NDOMIZE USR m1: OPEN #3,"lp": RA
NDOMIZE USR m1: OPEN #4,"dd": RA
NDOMIZE USR m1: POKE VAL "16096"
,a: RANDOMIZE USR m1: POKE VAL "
16090",VAL "136": RANDOMIZE USR
m1: POKE VAL "16092",b*oj: RANDO
MIZE USR m1: POKE VAL "16094",oo
: LPRINT
2068 CLS : PRINT AT oj,oh;"Rt Ma
rgin Justify?"TAB oh;"<Y> Yes
or <N> No": GO SUB sq: GO SUB lk
: LET jy=z: PRINT AT oj,oh;"Firs
t Line Indent?"TAB oh;"<Y> Yes
or <N> No": GO SUB sq: GO SUB l
k: LET nz=-z+ob

```

** The FIRST ON Initialization **

When Daisy is first loaded these prompts allow you to select Your type of Parallel Centronic Interface, and then initializes the LKDOS system for that. The software assumes a DUMB printer, and sets up LKDOS for the MAXIMUM Printer Line. All printers may be set up for either a LINE FEED as each carriage return occurs, or NO LINE FEED. The user group is split about 50/50 as to their Line Feed Choice of printer setting. This prompt allows the use of the program with whatever printer setting is the user's standard operating mode. One can operate Daisy with MANUAL printer settings by SELECTING DAISY WHEEL PRINTER at the prompts that follow during Initialization. This disables the "Dot Matrix Printer Control Section" that will be described later in the manual, and allows printing to be accomplished using MANUAL printer settings for the PRINT STYLE. The Dot Matrix Printer Control sections provides AUTOMATIC PRINTER STYLE CHANGES via MENU ELECTIVES.

The programming for these style changes are based upon EPSON COMPATABLE PRINTERS. If your printer is different, and does not respond correctly to Automatic Style Changes selected at menu, then while waiting for the opportune time to revise the programming, just select DAISY WHEEL during the initialization prompts. Then you can use the front panel switches of the printer to select the style of print desired. When operating in the Daisy Wheel Mode, the STYLE MENU is still

```

*****
D A I S Y
The TS-2068 Software System

For The

Larken Disk System

Copyright 1989 by:
Gulf Micro Electronics
1317 Stratford Ave.
Panama City, FL 32404
(rights reserved)
*
```

Select the INTERFACE
That you use:

- <1> Ollger or AERCO
- <2> Tasman CPI
- <3> A & J CPI

2067

used for the mathematics calculations for MATCHING LINE. Read the section that follow about FORMATTING and the STYLE MENU.

After the LKDOS is initialized, this section of programming is automatically DELETED from the main Daisy program. This "MERGE, USE, and DISCARD" concept is used throughout the Daisy Software to conserve memory for DATA Management.

YOU CAN RE-SET the Initialization ANYTIME during operation, without losing any data. You do this by <BREAK IN with the BREAK KEY and type: <GO TO 2060 ENTER>. There is another Re-initialization offered at the OFFICE TOOLS menu, but that one CLEARS the data out and loads in the standard Variable Files for a clean program.

*IMPORTANT FOR PRINTER
OPERATION*


```

*** Program "Inpt.B6" LLIST *** 3793 Bytes

150 RANDOMIZE USR VAL "24355"
158 LET a=PEEK 24314: LET b=PEEK 24313: LET c=PEEK
K 23560: LET a1=PEEK 24311+256*PEEK 24312
160 ON ERR GO TO VAL "252": IF c=12 THEN GO TO
370
162 IF c=13 THEN GO TO 150
164 GO SUB 365
166 GO TO (c=96)*386+(c=7)*310+(c=8)*290+(c=9)*27
5+(c=10)*265+(c=11)*255+(c=12)*370+(c=13)*150+(c=1
4)*392+(c=7)*392
235 POKE 23611,220
238 IF PEEK 23611<221 THEN GO TO 238
240 LET c=PEEK 23560: POKE 23611,220: IF c>31 AND
c<>96 THEN GO TO 300
250 GO TO 166
252 ON ERR RESET : GO TO 150
255 LET a=a-1: LET a1=a1-32: IF a<0 THEN LET a=0
: IF a1<b THEN LET a1=b
256 GO TO 301
265 LET a=a+1: LET a1=a1+32: IF a1+64>LEN u$ THEN
BEEP VAL ".2",VAL "14": RANDOMIZE USR VAL "24355
"
266 IF a>13 THEN LET a=14:
267 IF LEN u$<a1-b+1-a*32+481 THEN PRINT AT 0,0;
u$(a1-b+1-a*32 TO );
268 IF a<14 AND LEN u$>a1-b+1-a*32+480 THEN PRIN
T AT 0,0;u$(a1+1-b-a*32 TO a1-b+1-a*32+480);
269 IF a>13 AND LEN u$>a1-b+1-a*32+480+64 THEN P
RINT AT 0,0;u$(a1+1-b-a*32 TO a1+1-b-a*32+480+64);

270 PRINT AT a,b; OVER 1;" "; GO TO 235
275 LET a1=a1+1: LET b=b+1: IF b=32 THEN LET b=0
: LET a=a+1: IF a>21 THEN LET a=a-1: LET b=31: LE
T a1=a1-1
280 IF A1>LEN u$ THEN LET A1=A1-1: LET B=B-0a: I
F B=-1 THEN LET B=31: LET A=A-1
285 PRINT AT a,b; OVER 1;" "; GO TO 235
290 LET b=b-1: LET a1=a1-1: IF b=-1 THEN LET b=3
1: LET a=a-1: IF a=-1 THEN LET b=0: LET a=0: LET
a1=a1+1
295 PRINT AT a,b; OVER 1;" "; GO TO 235
300 PRINT AT a,b;CHR$ c;: LET u$=u$( TO a1)+CHR$
c+u$(a1+1 TO ): LET b=b+1: LET a1=a1+1: IF b=32 TH
EN LET b=0: LET a=a+1
301 IF LEN u$<a1-b+1-a*32+481 THEN PRINT AT 0,0;
u$(a1-b+1-a*32 TO )
302 IF LEN u$>a1-b+1-a*32+480 THEN PRINT AT 0,0;
u$(a1-b+1-a*32 TO a1-b+1-a*32+480)
305 PRINT AT a,b; OVER 1;" "; GO TO 235
310 LET d1=(1 AND d1<1)+(2 AND d1>0): LET d2=(a1
AND d1<2)+(d2 AND d1>1): IF d1=2 AND d2<a1 THEN L
ET I$=u$(d2+1 TO a1+1): LET u$=u$( TO d2)+u$(a1+2
TO ): LET d2=0: LET d1=0: LET a4=1: GO TO VAL "390
"
312 IF d1=2 AND d2<a1 THEN LET d1=0: LET a4=1: L
ET d2=0: GO TO VAL "390"
335 IF d1=1 THEN PRINT #0;AT 0,0; INVERSE 1;"Pos
ition Cursor to BLOCK END, Then EDIT": PRINT AT
a,b; OVER 1; FLASH 1;"_";: BEEP VAL ".2",15: GO TO
VAL "235"
340 IF d1=2 THEN LET a4=1: GO TO VAL "390"
365 PRINT AT 18,0; INK 5;"ENTER to Resume - Capst
SYM=HELP *Bik Insert - EDIT=Bik Delete. "; INK 7;
AT a,b; INVERSE 1; OVER 1;"_";: RETURN
370 PRINT AT a,b;" ";
371 IF a1>0 THEN LET u$=u$( TO a1-1)+u$(a1+1 TO
): IF a1>LEN u$ THEN GO TO 150
372 LET b=b-1: LET a1=a1-1: IF b<0 THEN LET a=a-
1: LET b=31: IF a<0 THEN LET a=0
373 IF a1<1 THEN LET a1=1: LET b=0
374 GO TO 301
386 PRINT INVERSE 1; OVER 1;AT a,b;" "; OVER 0;#0
AT 0,0;"BLOCK INSERT"; INVERSE 0;"<1> To Insert

```

This is The Whole Input And
Editing for DATA BASE
MANAGEMENT.

Accessed From The:

1. Function menu (INPUT EDIT)
2. QUICKIE MENU - Line 2926
3. The Sequential Files MANAGER
(Line 9668 - 9737)

During "Turbo I" Mode
This group "STAYS ATTACHED"
TO DAISY, TO AVOID WAITS
FOR MERGE.

Two Other Programs are
Attached in TURBO I
Mode:

1. EDT.B6 page 16
2. SEQT.B6 page 18

The Code Program "CAT.C1"
IS USED FOR FAST INPUT
TYPING AND RESIDES
IN ADDRESS 24355 + 266

ESCAPE BRANCHES BEGIN
AT LINE #392

Dont Be Intimidated!

Instead, let Daisy kinda "flow
into you" by dailly use and often
review of the manual. It isn't
difficult to get started if you read
enough of the manual to understand the
concepts. You will be able to do
simple operations that you are
accustomed to doing with other
software. Then everytime you use
Daisy you will learn something. And,
every time you review the manual you
will learn something else. This
process will go on almost forever. I,
the programmer, learn things from
using Daisy that were not planned as
functionall! Un-planned benefits occur
when so very many planned function
overlap in capability.

```

L$          <2> Insert Typing": PAUSE 0: L
ET a3=z: PAUSE 0: LET z=CODE INKEY$-48: LET a4=z:
LET z=a3: IF a4<1 OR a4>2 THEN GO TO VAL "386"
389 IF a4=1 THEN LET u$=u$( TO a1)+I$+u$(a1+1 TO
)
390 IF a4=1 THEN PRINT #0;AT 0,0;"
": GO SUB 365: GO TO 301
391 PRINT AT 18,0;"Type the Insert, OR ERASE Quot
esand enter String name to Insert.": INPUT LINE m
$: LET u$=u$( TO a1)+m$+u$(a1+1 TO ): PRINT AT 18,
0;"
": PRINT #0;AT 0,0;"
": GO TO VAL "301"
392 ON ERR GO TO 393: STOP
393 ON ERR RESET : CLS : IF sx=oo THEN PRINT #5
;u$: LET u$="": PRINT AT 0j,oo;"<1> More TEXT -or-
<2> QUIT": PAUSE NOT PI: LET qs=CODE INKEY$-VAL "
48": IF qs<oa OR qs>ob THEN GO TO VAL "393"
394 IF qs=oa THEN GO TO 150: REM GO SUB VAL "36
5": GO TO VAL "301": IF sx=1 THEN GO TO 9698
395 IF qs=ob THEN GO TO VAL "9687"
396 IF sx=oa THEN GO TO VAL "9698"
398 IF seq=oa THEN GO TO a1t
400 GO TO (xy<oa)*VAL "2950"+(xy=oa)*VAL "3028"+(
xy>oa)*ed

```

*** Program "edt.B6" LLIST *** 3061 Bytes

```

2926 ON ERR RESET : RANDOMIZE USR m1: LOAD "ed.C1
"SCREEN$ : LET d1=oo: LET sx=oc: LET w$="": LET s$
="": LET xy=ob: LET ep=(ep AND ep>oo AND LEN h$(oa
)>oc)+(og AND ep<oa AND LEN h$(oa)>oc)+(ep AND ep<
oa AND LEN h$(oa)<od): PRINT AT op,oi;LEN u$;" ":
AT op,t3-od;(LEN h$(oa) AND ep<oh)+(LEN I$(oa) AND
ep>og);AT oq,op; FREE ;" ":AT os,od;ep;AT ot,od;
eq
2928 GO SUB 11: GO TO (z>oj)*VAL "2928"+(z=oa)*VAL
"140"+(z=ob)*VAL "2930"+(z=oc)*VAL "3400"+(z=od)*
fm+(z=oe)*VAL "2966"+(z=of)*VAL "3010"+(z=og)*VAL
"3060"+(z=oh)*VAL "2968"+(z=oi)*VAL "2929"+(z=oo)*
VAL "2600"+(z=oj)*VAL "3040"
2929 POKE VAL "23658", (oo AND PEEK VAL "23658">oo)
+(oh AND PEEK VAL "23658"<oa): BEEP oa/od,on: GO T
O VAL "2928"
2930 RANDOMIZE USR m1: LOAD "prt.C1"SCREEN$ : LET
xy=oo: PRINT AT oc,ou; INK oe;"EDIT "; INK og: GO
SUB sq: PAUSE oo: LET z=CODE INKEY$-VAL "48": IF z
<oa OR z>ou THEN GO TO ed
2932 IF z<>op THEN LET s$=u$
2933 LET z1=z
2934 IF z1<oh THEN LET u$=h$(z1)
2935 IF z1>og AND z1<op THEN LET u$=I$(z1-og)
2940 IF z1=og THEN LET u$=a$
2941 IF z1=or THEN LET u$=b$
2942 IF z1=os THEN LET u$=c$
2943 IF z1=ot THEN LET u$=d$
2944 IF z1=ou THEN LET u$=e$
2945 CLS : LET a=oo: LET b=oo: LET a1=oo: GO SUB V
AL "365": GO TO m3+oa
2950 IF z1<oh THEN LET h$(z1)=u$
2951 IF z>og AND z1<op THEN LET I$(z1-og)=u$
2954 IF z1=og THEN LET a$=u$
2956 IF z1=or THEN LET b$=u$
2958 IF z1=os THEN LET c$=u$
2960 IF z1=ot THEN LET d$=u$
2962 IF z1=ou THEN LET e$=u$
2963 IF z1<>op THEN LET u$=s$
2964 LET s$=""
2965 GO TO ed
2966 LET u$="": GO TO m1+t5
2968 CLS : PRINT AT ou,oo;"Input L$ Data to Insert
Anywhere" during EDIT or Input Typing.": INPUT I
$: INVERSE oo: GO TO ed

```

*"Inpt.B6" section is use
for both MAIN TEXT DATA
Bases. Sequential Files
Manager and SMART Text
db Manager.*

The END BRANCH routines for the INPUT - EDIT group begins at line #392 which resets ON ERR, then Branches to return to the Data Base Manager which has been using the Input Edit functions.

*This is the EDIT section for the
Two DATA Bases Hb and I\$
ARRAYS. STAYS ATTACHED IF
TURBO=1*

Quickie Menu

1. Typing 2. Re-ed 3. Store
4. HELP 5. DELETE BUFFER
6. View/Edit Array Cells
7. Word Replace, Global
8. Input to L\$ for PASTE-UP
9. Cycle Caps 0 Edit Seq DB
- <: Zap a Data Base

Buffer= Cell Limit=
FREE=

H-0
I-0 ed

Slct Data to EDIT

<1> H\$(1)	<8> I\$(1)
<2> H\$(2)	<9> I\$(2)
<3> H\$(3)	10> I\$(3)
<4> H\$(4)	11> I\$(4)
<5> H\$(5)	12> I\$(5)
<6> H\$(6)	13> I\$(6)
<7> H\$(7)	14> I\$(7)

15> Typing	18> C\$
16> A\$	19> D\$
17> B\$	20> E\$

21> Abort
Enter Choice:

```

3010 RANDOMIZE USR m1: LOAD "ve,C1"SCREEN$: LET s
x=oc: LET qs=oc: LET seq=oo: LET xy=oa
3011 GO SUB sq: GO SUB 11: LET z1=z: IF z1>oc THEN
GO TO (z=od)*ed+(z>od)*fm
3012 IF z1<oc THEN GO TO VAL "3016"
3013 LET z1=ob: BEEP P1/oh,on: INPUT "Enter Data D
isk #";dd: RANDOMIZE USR m1: GO TO dd: CLS : RANDO
MIZE USR m1: CAT ".A$": BEEP P1/oh,os: INPUT "Ent
er NBR only of "; FLASH oa;"t"; FLASH oo;" File to
LOAD"; LINE m$: LET m$="t"+m$+".A$": RANDOMIZE US
R m1: LOAD m$ DATA i$( ): BEEP P1/oh,on: INPUT "Ent
er PGM Disk #";dd: RANDOMIZE USR m1: GO TO dd
3014 CLS : PRINT AT oj,oo;"The Data is now a ""|""
DB FILE and is ready for EDIT""After EDIT you
may RE-SAVE via Menu with an appropriate File #"
""ENTER WHEN READY For EDIT": PAUSE oo: CLS
3016 LET a=(LEN h$(oa) AND z1<ob)+(LEN i$(1) AND z
1>oa): IF a<ob THEN CLS : PRINT #oo;"No Data in t
he Data Base": BEEP ob,on: GO TO ed
3017 BEEP P1/oh,on: PRINT #oo;"Input Start Cell #
of "; FLASH oa;"H" AND z<ob)+( "I" AND z>oa); FLAS
H oo;" DB for View EDIT (1 to 7) ": INVERSE oo: P
AUSE oo: LET y=CODE INKEY$-VAL "48": CLS : IF y<oa
OR y>og THEN GO TO VAL "3015"
3018 FOR n=y TO og: CLS : IF z1=oa THEN PRINT AT
oo,oo;h$(n);
3019 IF z1=ob THEN PRINT AT oo,oo;i$(n)
3020 PRINT #oo;AT oo,oo: FLASH oa;"I" AND z1>oa)+
("H" AND z1<ob);("n;n"); FLASH oo: INVERSE oa;"E
NT for NEXT- or-"; INVERSE oo;" "; FLASH oa;"e to
EDIT"; FLASH oo: PAUSE oo: LET t$=INKEY$: IF t$<>"
e" AND t$<>"E" THEN GO TO VAL "3026"
3021 CLS : LET s$=u$: IF z1=oa THEN LET u$=h$(n)
3022 CLS : IF z1=ob THEN LET u$=i$(n)
3024 LET a=oo: LET b=oo: LET al=oo: LET d1=oo: GO
SUB VAL "365": GO TO VAL "301"
3026 NEXT n: GO TO ed
3028 IF z1=oa THEN LET h$(n)=u$
3029 IF z1=ob THEN LET i$(n)=u$
3030 LET u$=s$: LET s$="": LET n=n-oa: NEXT n: GO
TO ed
3040 CLS : PRINT AT oj,ob;"Zap Which DB?""TAB ob;
"[1] H Data Base""TAB 2;""[2] I Data Base""TAB 2;
"[3] Both": GO SUB sq: GO SUB 11: IF z=oa THEN DI
M h$(oa): LET ep=oo: LET cx=oo
3042 IF z=ob THEN DIM i$(oa): LET cz=oo: LET eq=oo
3044 IF z=oc THEN DIM h$(oa): DIM i$(oa): LET cx=
oo: LET cz=oo: LET ep=oo: LET eq=oo
3046 GO TO ed
3400 CLS : IF cx=oo OR cz=oo AND ep>og THEN PRINT
AT om,oa;"Inpt Para Lgth for ";("H$ array" AND cx
=oo)+("I$ array" AND ep=oh AND cx>oo): LET mz=oo
3402 IF cx=oo THEN LET ep=oa: INPUT cx: IF FREE
-cx*og<k4 THEN LET ep=oo: LET cx=oo: GO TO k3+m4
3404 IF cx>oo AND ep>og AND cz=oo THEN LET eq=oa:
INPUT cz: IF FREE -cz*og<k4 THEN LET cz=oo: LET
eq=oo: GO TO k3+m4
3406 IF ep=oa THEN DIM h$(og,cx)
3407 IF ep=oh AND eq=oa THEN DIM i$(og,cz): LET e
q=oa
3408 IF ep<oh THEN LET h$(ep)=u$: LET ep=ep+oa: G
O TO VAL "3411"
3409 IF ep>og AND eq<oh THEN LET i$(eq)=u$: LET e
q=eq+oa
3411 IF eq<oh THEN LET u$=""
3414 GO TO ed
3416 CLS : PRINT AT oj,oo;"Select: ""|""<1> Daisy DB
Manager""|""<2> Sequential files Mgr": GO SUB 11:
IF z<oa OR z>ob THEN GO TO VAL "3416"
3417 LET alt=(ed AND z=oa)+(VAL "9668" AND z=ob):
IF turbo<>oa THEN LET turbo=oa: GO SUB VAL "495":
RANDOMIZE USR m1: MERGE "tb.B6": GO TO VAL "850"
3418 GO TO alt

```

View Edit Options

- [1] View Edit "H" DB (mem)
- [2] View Edit "I" DB (mem)
- [3] LOAD/Edit Seq Files
- [4] To Quickie Menu
- [5] To Function Menu

These various Edit Functions operate By individual Selections from The Quickie Menu. (Line 2926)
 <goto Ed> gets The Quickie Menu from a BREAK STATUS.

— Turbo mode selection sets Turbo 1

This is the Sequential Files DATA
BASE MANAGER. STAYS ATTACHED
IF TURBO = 1

Sequential File Manager

- [1] Create Seq File of Lines
- [2] Create Seq File of Paragraphs
- [3] View - Edit File
- [4] PRINT FILE
- [5] Return to Daisy Menu

The Sequential File
DATA BASE Manager is
supported by the
"INPT. B6" section
and the
"EDT. B6" section.

The word processor "WdPro.86"
is merged when <3> (Below)
is elected.

Once printout is finished "wdpro.B6" is deleted.

Seq File View- Edit- Print

- ```

[1] View a File
[2] Edit a Para File
[3] PRINT a Para File
[4] PRINT a Line File
[5] To Main Menu

```

```

0: GO TO dd: GO TO VAL "9668"
9708 RANDOMIZE USR m1: LOAD "seq.C1"SCREEN$
9709 PAUSE oo: LET y=CODE INKEY$-VAL "48": IF y>ob
 THEN GO TO (y=oc)*VAL "2604"+(y>oc)*VAL "9668"
9710 PAUSE oo: IF y=ob THEN GO TO VAL "9722"
9712 CLS : INPUT "INPUT Left Margin:";mar: RANDOMI
ZE USR m1: POKE VAL "16094",mar: LPRINT : INPUT "L
ines per Page?";ln: INPUT "Start print at Line #?"
;st: INPUT "Page #?";pg
9714 RANDOMIZE USR m1: OPEN #5,x$: FOR n=oa TO m6:
 LET u$="": ON ERR GO TO VAL "9718": INPUT #5; LI
NE u$: LPRINT u$: GO SUB sp
9716 NEXT n
9718 ON ERR RESET : ON ERR GO TO VAL "9720": STO
P
9720 ON ERR RESET : RANDOMIZE USR m1: CLOSE #5: G
O TO VAL "9668"
9722 RANDOMIZE USR m1: POKE VAL "16090",lx: RANDOM
IZE USR m1: POKE VAL "16094",oo: LPRINT : IF turbo
=oo THEN RANDOMIZE USR m1: MERGE "wdpro.B6"
9724 RANDOMIZE USR m1: OPEN #5,x$: FOR n=oa TO ol:
 ON ERR GO TO VAL "9730": INPUT #5; LINE w$: GO S
UB te: GO SUB sp
9726 NEXT n
9730 ON ERR RESET : ON ERR GO TO VAL "9732": STO
P
9732 ON ERR RESET : RANDOMIZE USR m1: CLOSE #5: I
F turbo=oo THEN DELETE VAL "3742",VAL "3812"
9734 GO TO VAL "9668"
9735 GO SUB VAL "365": GO TO VAL "301"
9736 RANDOMIZE USR 100: CLOSE #5: INPUT "Enter PRO
GRAM DISK #:";dd: RANDOMIZE USR m1: GO TO dd: GO T
O VAL "9668"
9866 DELETE VAL "9822",VAL "9864": GO TO fm

```

#### INFORMATION . . .

Printing of LINE Files is  
to your INPUT TAB Position.

Printing of PARAGRAPH Files  
is to the PRE-SET FORMAT of  
the Word Processor.

You may Abort to set the  
Format Desired. . .

<1> Print Line File  
<2> Print Paragraph File  
<3> To Daisy Function Menu  
<4> To Seq Menu

ENTER CHOICE

\*\*\* Program "repp.B6" LLIST \*\*\* 1466 Bytes

```

2100 RANDOMIZE USR m1: LOAD "rp.C1"SCREEN$: LET p
c=oo: LET az=oo: LET no=oo: LET z1=oo: LET rpr=oa:
 LET pt=oo: LET ms=oo: LET mx=oo: LET ln=oo: LET g
g=oa: LET mm=oo
2101 GO SUB sq: GO SUB ll: GO TO (z<oa OR z>oh)*VA
L "2101"+(z>oo AND z<od)*VAL "2102"+(z=od)*fm+(z=0
f)*k2+(z=oe)*VAL "2174"+(z=og)*VAL "8990"+(z=oh)*V
AL "2149"
2102 IF z<oc AND turbo=oo THEN RANDOMIZE USR m1:
MERGE "usrpgm.B6": GO TO VAL "2103"
2103 IF z=ob THEN RANDOMIZE USR m1: MERGE "lh.B6"
: GO TO VAL "2104"
2104 IF z>oa THEN INPUT "Key: <1> for Manual Addr
essing <2> for Mail List Addr";mm: CLS : IF z
=ob THEN GO SUB VAL "2590": GO SUB VAL "4055"
2105 IF z=oc THEN CLS : PRINT AT os,ob;"Do you wa
nt Auto Page Control? <1> yes or <2> No ": INPUT
pc: CLS : PRINT AT os,ob;"Twin Labels? <1> Yes - <
2> No": INPUT tw: CLS
2106 IF z=oa OR mm=oa THEN INPUT "ENT nbr copies
";no: LET ms=oa: LET mx=no: IF mm=oa AND z=oc THEN
 GO SUB VAL "2170"
2108 IF z=oc THEN INPUT "ENT TAB ";tz: INPUT "ENT
sps";sx: CLS
2110 IF mm=ob THEN INPUT "ENT ml strt nbr ";ms: I
NPUT "ENT end ml nbr ";mx: CLS
2112 FOR l=ms TO mx
2116 IF z=ob THEN GO SUB VAL "4061": CLS
2118 IF z=oc THEN GO SUB VAL "2150"
2120 IF z<oc THEN GO TO VAL "2180"
2138 NEXT l
2139 LET lh=oo
2140 IF zm<mz THEN LET zm=zm+oa: GO SUB k9+oj: GO
TO VAL "2112"
2144 LET z2=oo
2148 GO TO VAL "2410"
2149 GO SUB VAL "8150": LET gg=oo: GO TO VAL "2410
"
2150

```

The MAIN PRINTING MANAGEMENT  
Group. STAYS ATTACHED IF TURBO=2  
Merges IF TURBO=1 OR 0.  
Two other groups STAY ATTACHED IF  
TURBO = 2 -  
<1> usrpgm.B6 <2> wdpro.B6

#### Printing Menu

<1> Manuscript  
<2> Letters/Invoices  
<3> Labels/Envelopes

<4> To Function Menu  
<5> Postscript ON  
<6> To Mailing List

<7> Print Disk Files  
<8> Print Outline  
<9> Print Disk Catalog

<0> Print Screens  
<:> Utilities Menu

2100

```

2152 IF mm=oa THEN LPRINT TAB tz;p$(oa)'TAB tz;p$
(ob)'TAB tz;p$(oc)'TAB tz;p$(od): LET nn=nn+od: LE
T a=oo: GO SUB VAL "2164"
2154 IF mm=ob THEN FOR n=oa TO od: LPRINT TAB tz;
o$(1,n)': NEXT n: LET nn=nn+od: LET a=oo: GO SUB V
AL "2164"
2160 IF tw=oa THEN LPRINT TAB tz;n$(of)'TAB tz;n$
(og)'TAB tz;n$(oh)': LET nn=nn+od: LET a=oo: GO S
UB VAL "2164"
2161 IF pc=oa AND nn>=ln-od THEN GO SUB pe
2162 RETURN
2164 IF a<sx THEN LPRINT ': LET a=a+oa: IF pc=oa
THEN LET nn=nn+oa
2166 IF a<sx THEN GO TO VAL "2164"
2167 IF pc=oa AND nn>=ln-od THEN GO SUB pe
2168 LET a=oo: IF pc<>oa THEN LET nn=oo
2169 RETURN
2170 DIM p$(od,VAL "26"): INPUT "First Line: ";p$(
oa): INPUT "Second line: ";p$(ob): INPUT "Third Li
ne: ";p$(oc): INPUT "Fourth Line: ";p$(od): RETURN
2172 DIM p$(oa): RETURN
2174 BEEP oa/ob,ov: LET scp=oa: GO TO VAL "2101"

```

THE Built-in MAIL LIST PRINTER: Line 2150 begins this utility. The Mailing List Data Base is printed via selection <3> at the Print Management Menu. Mail List records are 7 lines. All 7 lines of each of the records may be printed in a File Listing, or the first four lines may be printed for labels or envelopes. Such print products may be "of a memory file", or "of disk files" when "<9> Print Disk Files" has been selected at the Print Management Menu. In the latter case, the file numbers of selected disk files are automatically loaded and printed as either lists, to labels, or to envelopes.

\*\*\* Program "usrpgm.B6" LLIST \*\*\*

*998 Bytes / USER Programming Group. STAYS ATTACHED IF Turbo = 2.*

```

2182 LET w$=h$(1): GO SUB tel: GO SUB sp
2184 LET w$=h$(2): GO SUB tel: GO SUB sp
2187 LET w$=h$(3): GO SUB tel: GO SUB sp
2188 LET w$=h$(4): GO SUB tel: GO SUB sp
2190 LET w$=h$(5): GO SUB tel: GO SUB sp
2192 LET w$=h$(6): GO SUB tel: GO SUB sp
2194 LET w$=h$(7): GO SUB tel: GO SUB sp
2196 IF zy<oa THEN GO TO 2206
2197 REM * Disk Sequencer * Do not disturb lines
2198 - 2206
2198 IF zy>oa AND zx<zy THEN LET zx=zx+oa: LET az
=oo: LET m$=STR$ zx: LET m$=m$+".A$": RANDOMIZE US
R m1: GO TO dd: RANDOMIZE USR m1: LOAD m$ DATA h$(
): RANDOMIZE USR m1: GO TO pd: GO TO VAL "2182"
2200 IF az=oa THEN LET m$=STR$ zp+".A$": RANDOMIZ
E USR m1: GO TO dd: RANDOMIZE USR m1: LOAD m$ DATA
h$(): RANDOMIZE USR m1: GO TO pd: LET az=oo: LET
zx=zp
2204 LET az=oo
2206 ON ERR GO TO VAL "2280": IF I$(oa,oe)>"" THE
N ON ERR GO TO VAL "2208": STOP
2208 ON ERR RESET
2210 LET w$=i$(1): GO SUB tel: GO SUB sp
2212 LET w$=i$(2): GO SUB tel: GO SUB sp
2214 LET w$=i$(3): GO SUB tel: GO SUB sp
2216 LET w$=i$(4): GO SUB tel: GO SUB sp
2218 LET w$=i$(5): GO SUB tel: GO SUB sp
2220 LET w$=i$(6): GO SUB tel: GO SUB sp
2222 LET w$=i$(7): GO SUB tel: GO SUB sp
2280 ON ERR RESET
2300 IF lh>oo THEN GO SUB VAL "4090": GO SUB pe:
GO TO VAL "2138"
2399 REM ** END of User Program Group. Do not dis
turb AFTER.
2400 IF spl=oa THEN GO SUB pe: LET nn=oo: GO TO V
AL "2138"
2401 IF lh>oo THEN LET z=ob
2402 IF z=ob THEN GO SUB k4++9: LET z2=z2+oa: IF
lh>oo THEN LET z=oa
2403 GO SUB pe: LET nn=oo: GO TO VAL "2138"

```

*Schedules The Printing of The "H AND I" data Bases.*

*Edit These Lines to Schedule Other DATA PRINTING. Sequence AS desired*

*← The Disk Sequencer. Loads in Disk Files for Printing IF "Print Disk Files" selected. Loops BACK TO Line 2182 To Keep on Printing Disk Files. - Ignored IF "Disk Files Printing" NOT Selected.*

*Change print Sequence AS desired OR Add other PRINTING.*

*End Routines - This whole group OPERATES under control of The "L Counter" Line #2112 of The "repp.B6" Program.*



\*\*\* Program "wdpro.B6" LLIST \*\*\*

1371 Bytes

# The Daisy Word Processor

```

3742 LET nt=nz: IF nt=1 THEN LET ll=ll-5: LET tb=
tb+5
3743 IF LEN w$<4 THEN GO TO 3788
3744 IF w$(1 TO 3)=" " THEN GO TO 3810
3745 IF LEN w$<=11 AND w$(LEN w$-2 TO LEN w$)="
" THEN LET w$=w$(TO LEN w$-2): GO TO 3748
3746 IF LEN w$<=11 THEN GO TO 3748
3747 IF LEN w$>=11 AND w$(11-2 TO 11)=" " THEN
LET w$=w$(TO 11-2)
3748 IF w$(1)=" " THEN LET w$=w$(2 TO): GO TO 37
46
3749 IF LEN w$<=11 THEN GO TO 3800
3751 LET m$=w$(TO 11)
3753 IF m$(LEN m$)<>" " AND w$(LEN m$+1)=" " THEN
LET w$=w$(LEN m$+1 TO): GO TO 3758
3754 IF m$(LEN m$)<>" " AND w$(LEN m$+1)<>" " THEN
LET m$=m$(TO LEN m$-1): GO TO 3754
3756 LET w$=w$(LEN m$+1 TO)
3758 IF m$(LEN m$)=" " THEN LET m$=m$(TO LEN m$-
1): GO TO 3758
3759 IF jy=oa THEN GO SUB 3774
3760 LPRINT TAB tb;m$: IF nt=1 THEN LET nt=0: LET
ll=ll+5: LET tb=tb-5
3763 LET nn=nn+1: IF sd=2 THEN GO SUB sp
3764 IF nn>=ln THEN GO SUB pe
3768 GO TO 3743
3774 LET pp=LEN m$: IF pp=11 THEN RETURN
3776 FOR r=1 TO pp: IF m$(r)=" " THEN GO TO 3780
3778 NEXT r: GO TO 3774
3780 LET m$=m$(TO r)+" "+m$(r+1 TO pp): LET pp=LE
N m$: IF pp<>11 THEN LET r=r+1
3784 IF pp<>11 THEN GO TO 3778
3786 RETURN
3788 IF LEN w$<3 THEN GO TO 3810
3790 IF w$(1 TO 3)=" " THEN GO TO 3810
3792 IF w$(1)=" " THEN LET w$=w$(2 TO): GO TO 37
92
3794 IF LEN w$<=11 THEN GO TO 3800
3795 IF w$(LEN w$)=" " THEN LET w$=w$(TO LEN w$-
1): GO TO 3794
3796 GO TO 3751
3800 LPRINT TAB tb;w$: IF nt=1 THEN LET ll=ll+5:
LET tb=tb-5: LET nt=0
3802 LET nn=nn+1: IF sd=2 THEN GO SUB sp
3804 IF nn>=ln THEN GO SUB pe
3810 LET w$="": IF nt=1 THEN LET ll=ll+5: LET tb=
tb-5
3812 IF gg=oa THEN RETURN

```

\*\*\* Program "pe.B6" LLIST \*\*\*

541 Bytes

```

5900 IF pr=oa THEN LPRINT "":TAB qq;pg;"
": LET nn=oa: LET pg=pg+oa: RETURN
5901 IF nn<ln+ob THEN LPRINT: LET nn=nn+oa: IF n
n<ln+ob THEN GO TO VAL "5901"
5902 IF cp>ob THEN GO TO VAL "5906"
5903 LPRINT TAB qq;pg: LET nn=oa: LET pg=pg+oa: OU
T po,ol: IF pr=oa THEN GO TO VAL "5913"
5904 GO SUB VAL "5912": GO TO VAL "5913"
5906 IF cp=ob THEN LPRINT TAB lx/ob;pg: LET pg=pg
+oa
5908 LET nn=oa: GO SUB VAL "5912"
5909 LET cp=(oa AND cp=ob)+(ob AND cp=oa): IF cp=o
a THEN LET qq=(VAL "34" AND pr=ob)+(VAL "36" AND
pr=oa): LET tb=oj
5910 IF cp=ob THEN LET qq=(VAL "92" AND pr=ob)+(V
AL "99" AND pr=oa): LET tb=(VAL "66" AND pr=ob)+(V
AL "71" AND pr=oa)
5911 GO TO VAL "5913"
5912 BEEP oa,ov: CLS: PRINT AT oj,ob;"Feed New Sh
eet, then ENTER": PAUSE oa: LET nn=oa: CLS: RETUR
N

```

The Daisy Word processor is a simple "line formatter-printer". (And most users think of Smart Text as a WORD PROCESSOR.). The Word processor will print anything that it sees "In W\$". You can even use it in the Direct Mode. Try this: <LET W\$="Keep it simple stupid!"> <GO TO TE ENTER>.

So, the key is what you put into "W\$" and send to the line #3742 via <GO TO or GO SUB te>.

The formatting is done via "variable switches" that turn ON or OFF certain functions, and assign Line Length (ll) and TAB (tb). These variable are managed by the Smart Text Menu selections. At line 3742, if nz=1 then you have elected "first line indent". Then the "1" is transferred to var "nt" and the line length is reduced by five characters (LL). Then the W\$ data is processed. Traps are to prevent EMPTY string from reaching the following formatter lines. Then traps prevents SHORTER than full lines of data from being processed. Any line shorter than line length "LL" is printed at line #3800.

Line 3748 trims off any spaces at the beginning of the W\$ data group. Line 3747 detects "Sentence structure" and then line 3749 sends a sentence line to line 3800 to be printed. Line 3751 begins to "nibble" W\$ into "line length streams of characters". Each nibble reduces W\$ by a line length and that data is placed into m\$. The processing of M\$ results in trimming of leading spaces, then trailing spaces. Then any "partial word" is trimmed off and put back into the main text (w\$). The result is a group of characters in m\$ that may not be a whole line of characters. If not, then 3759 evaluates "jy". If jy=1 then you have elected "right margin justify" and the m\$ characters are sent to be justified at the routine beginning at 3774.

Spaces are inserted between words to push the last character out to make LEN M\$=11. Then line 3760 prints the whole line of characters, beginning at TAB tb. If it is the "first line, and INDENT is in force, then indent is cancelled (LET nt=0:LET ll=ll+5:LET tb=tb+5). Line 3762 increments the line counter nn, and evaluates "sd". If sd=1 then Double Spacing is in effect and a line space is printed (GO SUB sp). Line 3764 evaluates for page length. If nn (line counter) is at the pre-set page length (ln) then a page end is directed by <GO SUB pe>. Line 3768 then returns to the beginning to nibble off another "line length" group of characters from the text carrier w\$.

The process continues until the last line of characters is LESS than ll in length, upon which time line 3746 sends the end line to line 3768 for printing. Then all data in w\$ has been printed, and w\$ has been eliminated. The clean-up routine follows at line 3802. Line 3812 evaluates "gg". If gg=1 then "sequenced printing" is in progress. A RETURN is to the User Program Group to pick up the next W\$ data group, upon which the W\$ group is again sent to line TE (3742) for printing. If sequenced printing is NOT in effect, then gg=0, and line #3814 executes to GO TO the function menu.

\*\*\* Program "lh.B6" LLIST \*\*\*

780 Bytes

```

4055 INPUT "ENT Date: ";n$(oj): IF mm=ob THEN RET
URN
4056 INPUT "1st Name ";n$(oe)
4058 CLS : PRINT AT oj,oo;"(Optional: Omit line w
th ENTER)"; INPUT "Co Name ";n$(oa): INPUT "Dept "
;n$(ob): INPUT "St adr ";n$(oc): INPUT "Cty,St,Zip
";n$(od): CLS
4059 IF gg=oa THEN RETURN
4060 LET nn=oo
4061 IF lh=oa AND pg<ob THEN FOR j=oa TO ok: GO
SUB sp: NEXT j: LPRINT TAB qq-ob;n$(oj): LET nn=nn
+oa: GO SUB sp: GO TO VAL "4073"
4062 FOR t=of TO oj
4064 IF n$(t)(oa)=" " THEN NEXT t
4066 LET m$=n$(t)
4067 IF m$(LEN m$)=" " THEN LET m$=m$(TO LEN m$-
oa): GO TO VAL "4067"
4068 LET pg=oa: IF m$(oa)<>" " THEN LPRINT TAB qq
-LEN m$/ob;m$: LET nn=nn+oa
4069 IF t=oj AND sp=oa THEN RETURN
4070 IF t=oj THEN GO SUB sp
4071 IF t=oj THEN GO TO VAL "4073"
4072 NEXT t
4073 LET a=oa
4074 IF mm<ob THEN LET m$=n$(a)
4076 IF mm=ob THEN LET m$=o$(1,a)
4078 IF a=oe THEN LET m$="Dear "+m$
4080 IF a=oe THEN GO SUB sp: LPRINT TAB tb;m$: LE
T nn=nn+oa: GO SUB sp: IF gg=oa THEN RETURN
4082 IF a=oe THEN GO TO VAL "3956"
4084 IF m$(oa)=" " THEN LET a=a+oa: GO TO VAL "40
74"
4086 IF m$(oa)<>" " THEN LPRINT TAB tb;m$: LET nn
=nn+oa
4087 LET a=a+oa
4088 GO TO VAL "4074"

```

Letter Support Group  
Merges AS Needed.  
Handles either MANUAL input  
OR MAILING LIST Merge of  
NAMES, ADDRESSES.  
FORMATS And PRINTS letterhead,  
Business address Block,  
Salutation.

| FORMAT MENU           |                  |       |       |
|-----------------------|------------------|-------|-------|
| Select until Finished |                  |       |       |
| <1>                   | Single Spc       |       |       |
| <2>                   | Dbl Spc          |       |       |
| <3>                   | Print Style Menu |       |       |
| <4>                   | Block Indent Off |       |       |
| <5>                   | Block Indent ON  |       |       |
| <6>                   | To Fun Menu      |       |       |
| <7>                   | Column Print     |       |       |
| <8>                   | Let Head Strny   |       |       |
|                       |                  | 2083  |       |
|                       | Pg Cent=         | Pg    | Lgth= |
| Max                   | Line             | Match | Line= |
| Line                  | Lgth             | Line  | Page= |
|                       |                  |       | Tab = |

\*\*\* Program "fomat.B6" LLIST \*\*\*

1772 Bytes

```

2080 RANDOMIZE USR m1: LOAD "fo5.C1"SCREEN$: LET
mat=11: LET xo=ix: LET lo=11: LET rs=ld: LET cp=oo
: LET lh=oo
2081 PRINT AT ol,VAL "24";"ld=";ld;" " ;AT oe,os;(
"a" AND sd<ob)+(" " AND sd>oa);AT of,os;"a" AND s
d>oa)+(" " AND sd<ob);AT op,ob;"Pica " AND ps<oc)
+"Elite" AND ps=oc)+("Cond " AND ps>oc);AT ot,ob;
lx;AT op,oq;qq;AT op,VAL "28";ln;AT ot,ol;ll;AT ot
,oq;mat;AT or,VAL "28";nn;AT os,VAL "28";pg;AT ot,
VAL "28";tb: GO SUB sq: GO SUB ll: IF z<oa OR z>oh
THEN GO TO VAL "2081"
2082 IF z=og THEN LET ps=od: LET lx=(VAL "126" AN
D pr>oa)+(VAL "136" AND pr<ob): LET ll=(VAL "50" A
ND pr>oa)+(VAL "55" AND pr<ob): CLS : PRINT AT oj,
ob;"Start Col 1 or 2 ?": INPUT "key 1 or 2";cp: IF
cp=oa THEN LET qq=(VAL "34" AND pr>oa)+(VAL "36
AND pr<ob"): LET tb=oj
2083 IF z=og AND cp=ob THEN LET qq=(VAL "90" AND
pr=ob)+(VAL "99" AND pr=oa): LET tb=(VAL "64" AND
pr=ob)+(VAL "70" AND pr=oa)
2084 LET sd=(z AND z<oc)+(sd AND z>ob): IF z=od TH
EN LET tb=(tb-oe AND ll<lx-ol)+(tb AND ll>lx-oj):
LET ll=(ll+oj AND ll<lx-ol)+(ll AND ll>lx-oj): LE
T ld=ld-oj: BEEP oa/od,ol: PRINT AT ot,ol;ll;AT ot
,oq;mat;AT ot,VAL "28";tb;" " : GO TO VAL "2081"
2085 IF z=oe THEN LET tb=(tb+oe AND ll>ou)+(tb AN
D ll<VAL "21"): LET ll=(ll-oj AND ll>ou)+(ll AND l
l<VAL "21"): LET ld=ld+oj: BEEP oa/od,ol: PRINT AT
ot,ol;ll;AT ot,oq;mat;AT ot,VAL "28";tb;" " : GO T
O VAL "2081"
2086 GO TO (z<oc)*VAL "2081"+(z=oc)*VAL "2093"+(z>
oc)*VAL "2088"
2088 BEEP oa/od,os: LET lh=(oa AND z=oh)+(oo AND z
<>oh): IF cp=oa AND pr=VAL "1" THEN LET lx=(VAL "
134" AND ps>VAL "3")+ (VAL "96" AND ps=VAL "3")+ (VA

```

The FORMAT SECTION, lines 2080-2081, MERGES when FORMAT is selected at the Function Menu. This group MERGES in the Style Menu if needed for Printer Style Changes (lines 8001 thru 8110). The FORMAT MENU (above) displays all of the needed information about Print Style, and the existing Printing Format. The numbered electives allow for a plethora of FORMAT changes. The PROMPTS after allow the standard DAILY CENTER PRINT format, or OFFSET printing, or Column Print. BLOCK INDENT <ON-OFF> selection allows text to be printed "Indented 5 characters" or "out-dented five characters" from both margins. COLUMN PRINT selection allow the print to start at either Column 1 or Column 2, and at top of page or any line number down from the top. The "PAGE END" routine does the "column switching". The Column Print Mode is most valuable when using Cut Sheets of paper to print two formatted columns.



```

L "80" AND ps<VAL "3"): REM Dot Matrix
2089 IF cp<oa AND pr=VAL "2" THEN LET ix=(VAL "12
4" AND ps>VAL "3")+(VAL "96" AND ps=VAL "3")+(VAL
"80" AND ps<VAL "3"): LET qq=INT (ix/ob): REM Dais
y
2090 CLS : PRINT AT oj,ob;"INPUT Printer Line Spa
cing""<1> 1/8""=76 Lines""<2> 11/64""=56 Line
s""<3> 3/16""=50 Lines""<4> 1/4""=37 Lines": I
NPUT x: LET ln=INT (VAL ".6"+(VAL "9.5"/VAL ".125"
AND x<ob)+(VAL "9.5"/VAL ".172" AND x=ob)+(VAL "9
.5"/VAL ".187" AND x=oc)+(VAL "9.5"/VAL ".25" AND
x>oc)): CLS : PRINT AT oj,ob;"Set Printer for Line
Spacing""TAB ob;"Then Enter Lines per page?"'TA
B ob;"Current page lgth=";ln: INPUT ln: CLS : INPU
T "Enter Page#";pg: INPUT "Enter Line Count start"
;nn: IF cp>oo THEN GO TO VAL "2092"
2091 CLS : PRINT AT oj,oo;"Line length=";ll:"Matc
h Line=";mat': INPUT "New Line Length?";ll: LET q
q=INT (ix/ob): LET tb=INT ((ix-ll)/ob): PRINT "Tab
=";tb: INPUT "New Tab?";tb

```

\*\*\* Program "stymn.B6" LLIST \*\*\* 1899 Bytes

```

8001 RANDOMIZE USR ml: LOAD "ps.C1"SCREEN$
8002 LET lt=oo: LET px=ps: LET xo=lx: LET lo=ll: G
O SUB sq: GO SUB ll: IF z<oa OR z>og THEN GO TO k
3+ob
8004 IF z=og THEN GO TO VAL "8114"
8005 IF pr=ob AND z<oe THEN GO TO VAL "8018"
8006 IF pr=ob AND z>od THEN GO TO VAL "8002"
8008 IF z=of THEN LET ix=INT (ix/ob+oa/ob): LET q
q=ix/ob: GO SUB VAL "8016": LET ma=INT (oa/ob+lx*l
o/xo): GO SUB VAL "8014": LET tb=INT ((ix-ll)/ob+o
a/ob): BEEP oa/ob,ov: GO TO VAL "8114"
8010 IF z=oe THEN LET lt=oa: GO SUB VAL "8110": B
EEP oa/ob,ov: GO TO VAL "8114"
8012 LET ps=z: LET lx=(t8 AND ps<oc)+(t9+of AND ps
=oc)+(VAL "136" AND ps=od): LET qq=INT (ix/ob+oa/o
b): LET ma=INT (oa/ob+lx*lo/xo): GO SUB VAL "8014"
: LET tb=INT ((ix-ll)/ob+oa/ob): GO TO VAL "8024"
8014 CLS : PRINT AT oj,ob;"Key In Line Length""TA
B ob;lx;"=Max ";ma;"=Match Margins": INPUT "Inpu
t Line Length ";ll: CLS : RETURN
8016 GO SUB bu: OUT po,zo: GO SUB bu: OUT po,t8+og
: GO SUB bu: OUT po,oa: GO SUB bu: OUT po,VAL "155
": GO SUB bu: OUT po,t8+og: GO SUB bu: OUT po,oa:
GO SUB bu: OUT po,oy: RETURN
8018 LET ps=z: LET lx=(VAL "84" AND ps<oc)+(VAL "9
8" AND ps=oc)+(VAL "126" AND ps=od): LET qq=INT (l
x/ob+oa/ob): LET ma=INT (oa/ob+lx*lo/xo): GO SUB V
AL "8014": LET tb=INT ((ix-ll)/ob+oa/ob): GO TO VA
L "8114"
8024 GO SUB mt: GO TO VAL "8114"
8030 REM ** The Dot Matrix Printer Control **
8031 IF pr<>oa THEN RETURN
8034 GO SUB bu: OUT po,zo: GO SUB bu: OUT po,m1+ou
: GO SUB bu: OUT po,oo: GO SUB bu: OUT po,VAL "155
": GO SUB bu: OUT po,m1+ou: GO SUB bu: OUT po,oo:
GO SUB bu: OUT po,zo: GO SUB bu: OUT po,oe+t4: GO
SUB bu: OUT po,oo: GO SUB bu: OUT po,VAL "155": GO
SUB bu: OUT po,oe+t4: GO SUB bu: OUT po,oo: GO SU
B bu: OUT po,zo: GO SUB bu: OUT po,VAL "87": GO SU
B bu: OUT po,oo: GO SUB bu: OUT po,VAL "155": GO S
UB bu: OUT po,VAL "87": GO SUB bu: OUT po,oo: GO S
UB bu: OUT po,os: GO SUB bu: OUT po,VAL "146": GO
SUB bu: OUT po,zo: GO SUB bu: OUT po,oc+t5: GO SUB
bu: OUT po,VAL "155": GO SUB bu: OUT po,oc+t5: GO
SUB bu: OUT po,oy
8035 GO SUB bu: OUT po,zo: GO SUB bu: OUT po,t6+ol
: GO SUB bu: OUT po,VAL "155": GO SUB bu: OUT po,t
6+ol: GO SUB bu: OUT po,oy: GO SUB bu: OUT po,zo:
GO SUB bu: OUT po,t8: GO SUB bu: OUT po,VAL "155":
GO SUB bu: OUT po,t8: GO SUB bu: OUT po,oy
8036 IF lt=oa THEN GO SUB VAL "8110"

```

Print Style And Printer Control  
Section. Menus When Needed

Select Print Style

- <1> Standard Pica 80
- <2> Pica HI Qual 80
- <3> Elite HI Qual 96
- <4> Condensed 136
- <5> Italic
- <6> Expand width
- <7> Escape

8000

THE STYLE MENU GROUP, (above is menu), MERGES from the FORMAT MENU for printer Style Changes. When a printer style is changed via menu elective PROMPTS present "a Matching Line Length" and asks for input of "Desired Line Length" for the newly selected style. Then if DOT MATRIX Printer is in use, the DOT MATRIX PRINTER CONTROL GROUP at line #8034 is used to make the printer change its print style. Line #8034 and 8035 CANCELS all printer styles back to the printer's without resetting the TOP OF FORM. Lines 8036-8054 affects the style change at the printer. Line 8080 is a "PRINTER READY" check which loops itself if the printer is off line (busy). You should TURN PRINTER ON, then use its panel switch to place it OFF LINE. Then type: PRINT IN 127 ENTER. EDIT line #8080 and replace "253" with the number given from PRINT IN 127. Different TS-2068's give different results of IN 127.

The PRINT STYLE MENU and PRINTER CONTROL SECTION. The printer Control section, lines 8034-8110 is for EPSON COMPATABLE Dot Matrix Printers. Other type of printers will require EDIT and Adjustments of the control codes to cause the same effect with the printer. Consult Printer Manual for its CONTROL CODE COMMANDS.

Lines 8034 Through 8110  
are for DOT MATRIX Printer  
Auto STYLE Changes.

IF you have printer problems,  
Just Elect "DaisyWheel"  
During First of Prompts.  
This will set pr=2 and let you  
do printing with MANUAL set  
print styles. (While you ARE  
struggling with Editing these  
lines for your printer).

## **\*\* Introduction \*\***

Daisy is a large software ensemble for the TS-2068 which contains several separate, but integrated, softwares. The Daisy system makes use of the DOS MERGE function extensively in order to control more programming than can be installed in TS-2068 memory at one time. All together Daisy controls about 80K bytes of programming, most of which remains in disk until called for.

Three Softwares of the Daisy ensemble are:  
1. The Daisy Shell Manager. 2. The Smart Text Disk Data Base Manager and Word Processor. 3. The Mail Merge Software. Two more major softwares are Managed by Daisy but not supplied in the Daisy disk. These are Update Magazine's "D.Base-1" software and Bob Mitchell's "View Calc Disk".

In addition, the Daisy ensemble includes a group of Utility programs selectable from menu. These utility programs allow the user to print out the data files of other softwares such as Mscript, Tasword, Modem down load files, and generally any ASCII file that has been saved to disk.

This manual is intended to give you an overview of the Daisy Software, with enough detail to get you started creating and printing text data bases. The manual will only "touch the surface" of describing the total capabilities of Daisy. Your learning process will evolve as you use the software. Each new use will bring new discoveries for at least the next year. There is already quite a volume of documentation about Daisy given in articles in Update Magazine. Almost every issue of the magazine has at least one section of the Daisy software given as a separate operating program. We started in October 1987 with a complete listing and operational guide for the "Mail Merge Software", which is now a part of the Daisy ensemble.

Other articles in Update Magazine that give detailed operation and programming information about sections of the Daisy software are: "J-Utilities" Oct 87 issue, "List Looker" and "Purity" January 1988 issue, "MS-TAS" July 1988 issue, "Vars File SAVE LOAD" Jan 89. The program "TS-2068 Data Input" given in the April 89 issue of Update is the Input and Edit functions of Daisy's main Text Data Base. Then, the July 89 issue of Update also has several of the Daisy program sections given. These are titled "D.Base-1", "LKDOS TOOLS", "Universal ASCII File Printer", and "View Calc for Disk". So, almost every issue of Update Magazine is an annex of this Daisy Manual.

More sections of the Daisy software will be published in future issues, not to sell the Daisy software, but to give good programming to the readers. So, you can associate Daisy with the daughter's wedding, ie, "something old, something borrowed, and something new, all packed together, and with a big frosted cake. And like a marriage, Daisy requires some patience and learning to make it work. But that isn't a departure from other large softwares. Some are so complex that there are 2 Week schools on learning Lotus 1-2-3 or Word Perfect, both for the IBM PC. The last filer that I received quoted tuition for WP classes at \$450.00. So, let us approach the Daisy learning process with the understanding that it is in the

same class as those big softwares, and it won't be learned overnight.

But Daisy isn't as complicated to operate as those softwares mentioned, and I believe that with a bit of study and practice you'll be doing fairly good work with it in about a week's time. I worked hard to make Daisy a very "user friendly" software. But any programmer knows that the larger a program is the less friendly it becomes. It will help you to understand the concepts used in building this software, starting with the "Smart Text" annex of Daisy.

**MENU DRIVEN SOFTWARE:** Smart Text represents about seven years of on-going programming and updates. The Smart Text software "outgrew" the TS-2068's memory. So many functions were added that there was no FREE memory for DATA accumulation. Fortunately, Disk Drive systems were introduced for the TS-2068, which opened up new ways of programming. As with most softwares for the TS-2068, Smart Text originally used cryptic "command codes" which one had to either memorize or paste up on the wall for reminders of how to operate. Soon my computer room looked like a patch-work quilt with so many notes pasted everywhere. Now I ask you, if the programmer cannot remember all of them special key codes needed, then the user will likely have some problem, eh?

So, the first thing done with the DOS Versions of Smart Text was to change the concept to "MENU DRIVEN" software. All of the command codes were discarded and "subjective menus" used. Some 20 menus were created for Smart Text, and all SAVED TO DISK. These menus popped up from disk as needed for software operation. Daisy follows with about 30 menus in disk. The design concept is to "Present a subjective Menu when needed by the function that is in progress". Then each menu itself is informative. One does not need to refer to a crib sheet to find out that "Caps Shift while holding the Symbol shift and the Y.key" does a certain essential function.

As design work continued on the Daisy program, it too out-grew the TS-2068 Memory capacity, even after saving 18000 bytes of memory by using disk to store the menus. The next solution was to break the BIG program down into functional blocks and store the functions in disk. This concept involves leaving most of the software in Disk and MERGING sections in as needed.

So, while Daisy is already up to 80K in program length, it can still be made even larger, and the plan is to do just that. We will add more functions as the users ask for them, and keep on building until we run out of Ingenuity. Daisy now controls three other softwares and manages the data products of almost all existing software for the TS-2068.

For now, Daisy provides a complete range of ADMINISTRATIVE functions. Its data bases are so very flexible that one can hardly find a needed format that cannot be created. When you think about it, 90 per-cent of the uses for a computer are "administrative", involving Data Base Management and Printing. Daisy is designed to encompass that 90%, to be loaded once in the morning to accomplish almost all of your daily Needs.

## **\*\* The Daisy System \*\***

The name DAISY was selected to be non-descriptive for a reason. This Daisy.B1 software is really a "System". Daisy encompasses Data Base Management, Word processing, and Disk Management, much in the manner of a DOS, or CP/M. The Daisy System contains more programming than the whole TS-2068 Computer, or for that matter, more than most whole computers provide in their built in ROM memory. Most of the programming is in Basic, but there are some Machine Code utilities used for providing SPEED where speed is needed. Basic programming provides easy "user access" to the running programs. One can almost always break into the programs by use of the BREAK Key to investigate the status of the computer or to use the computer for direct operation from the keyboard. This "open software" concept has both advantages and disadvantages.

One wouldn't think that such a large software as Daisy is would be able to accommodate break-in to allow the user to accomplish direct use of the computer. There are times when such freedom is very valuable. A good example would be when one is typing in data to a document and needs to insert a figure that can only be gotten by calculation. With Daisy this is simple to do. One just BREAKS and types <PRINT 4936/62>. The answer (79.6) is presented on screen. Then <GO TO fm ENTER> will resume Daisy operation. One can even MERGE a utility to Daisy, use it, and either let the utility remain attached, or Delete it. When finished <GO TO fm ENTER> resumes Daisy operation. No other software, especially large softwares, allow such freedom of user options.

But there is a down side of allowing user freedom of operations. Since Basic Report codes are not suppressed, every mistake in operation of Daisy will cause a REPORT CODE. This may be disconcerting at first during the learning process, but Report Codes are helpful. A report code allows you to analyze your mistakes and thus to learn how to operate more efficiently. Report Code stoppages would be terrible if they destroyed a portion of the data or if you couldn't recoup and resume operation. Daisy provides easy access back into the program, via the simple command "<GO TO fm ENTER>". There is no possibility of losing data by making a mistake that causes a report code. The only mistake that would cause the loss of data would be typing <CLEAR> or <RUN>. So, the first thing to learn about Daisy is <GO TO fm ENTER> to resume operation.

DAISY IS B-I-G!! It is the largest and most comprehensive software ever created for Sinclair Computers. Daisy DWARFS all other software for the TS-2068, is much bigger and more comprehensive than the QL's QUILL, and is a more comprehensive software than IBM's Word Perfect (that sells for several hundred bucks). The total program length of the Daisy system is about 80K bytes. But Daisy reaches outside of its own environment to use several other softwares for the TS-2068, such as Mscript, Tasword, D.Base-1, and View Calc. Daisy is a "Shell Manager" of about 8K in program length which manages 20 other programs in disk and 30 Menus of functions. Within all of these menus there are more than 200 choices of functions. As functions are elected from menu Daisy MERGES in the appropriate program. Then when the function

has been accomplished, Daisy deletes the function and presents another menu, thus keeping FREE operating memory at maximum.

The MERGE concept used by Daisy is assisted by a "Turbo" mode selectable at menu to allow certain often used functions to remain attached to the Daisy Shell program, thus eliminating the short pauses needed to merge in programs from disk. The use of TURBO ON mode reduces FREE Memory to around 21K. The only advantage of using TURBO is for rapid switching between Data Base management and printing functions. This type of operation is typical of writing and printing several different letters at one sitting. The TURBO OFF mode allows very large data bases to be created in memory at a small sacrifice in speed of switching between functions.

The Daisy Data Bases: Daisy is a Data Base Manager having such a variety of flexible data bases as no other software for any computer can match. There are SIX data bases and each has branches. These flexible data bases cover all conceivable needs for creating files for any purpose. Daisy provides complete DISK MANAGEMENT for each of the data bases. As files are created and edited they are stored in disk with discrete file names. Disk files may be re-loaded from disk for after editing. Disk files may be loaded for printing by the Daisy Word processor. The disk files may be printed singly or chained and printed in a continuous print out. With a few key punches you can schedule the print out of a thousand page book which will be printed non-stop. Or, a few key punches will schedule a letter to be printed to all names in one mailing list or several mailing lists, literally thousands of letters all perfectly addressed and ready to be stuffed into envelopes.

The MAIN TEXT DATA BASE: The selection "<1> Create - EDIT" enters this mode to create and edit the main text data bases. This is a dual data base having two names. We simply call them the "H" and "I" data bases, named after their data carriers. The data carriers are two Character arrays "H\$( )" and "I\$( )". These two arrays are dimensioned to the user's needs via prompts. You can think of these two data bases as "Paragraph Groups" that are each designed to store Edited Paragraphs, their length to be set by the user. When the first data group has been created and you are ready to tuck it away into memory storage a prompt comes to INPUT LENGTH. If you input, say, 1000, then the data base is set up for SEVEN groups of data, each 1000 characters in length. This will be the "H" data base. After the prompt, the data group that you have created is STORED as the "First of Seven".

As you proceed to create more data, six more STORE actions (in the "H" Data Base) are available. Then as more data is created, the next "I" Data base is set up and filled in similar fashion, first by prompt for Size, and then the Store Actions follow. After 14 STORE Actions have been accomplished the two data bases "H" and "I" are full. Typically this is about 14K of total data or about 6-8 pages of data. The data is in MEMORY and can easily be recalled to screen for editing again and again until it is perfect. Once both the "H" and "I" data bases are filled with 14 Stored groups of data, then Management decisions

are necessary. You can use the menus to PRINT the data, or you can use the menus to SAVE to disk. The SAVE to disk is with a discrete "numbered file". The discrete numbered file will later be used by the management system for controlled LOAD of data files for either editing or print-out.

**THE MAIL MERGE DATA BASE:** Here we have a separate software "Mail Merge" that is selectable at menu to create, edit, and manage Mail List Files. The selection for MERGING the Mail Merge software is at the OFFICE TOOLS Menu. You can create Mail list files of about 50 names and addresses. These files may be viewed, corrected, sorted, or SAVED for future use. Also, the Mailing list may be printed as a listing, or to labels, envelopes, or used immediately for addressing individual letters or chains of letters as selected by the Daisy menu. Upon exit of the Mail Merge functions the program itself is automatically Deleted from the Daisy shell.

**THE OUTLINE DATA BASE:** This is selectable at the Office Tools Menu for creating outlines of a type used in school schedules, church activities, or illustrative writing. The data base is completely flexible and can be set up to the user's needs. The theme is Major topics and Minor topics "under the Majors". Prompts allow you to set up as many Major Topics, and as many "Minor topics of each Major" as you desire. The outline prints out under page control with optional TAB settings.

**"D.BASE-1" Data bases.** Here we use plural because D.Base-1 has its own variety of data bases that can be created and managed. We are talking about a data base that contains "Records". Some of the various data bases are; Inventory records, Mail File records, numeric records, monthly appointment schedules by day, daily diary, etc. D.Base-1 files may be loaded to Daisy for including in its print products, mixed with text, outlines, etc. While D.Base-1 is not supplied with Daisy, the Daisy menus coordinates its load and operation.

**SEQUENTIAL FILES Data Bases:** This pair of Data Bases are for the LKDOS users, as no other TS-2068 system has such capability. Imagine! You may have a combination of other data bases in memory, enough to max out the FREE memory available. You can still create 50K more of data without disturbing the data that is in memory. You do it with Sequential files sent directly to disk as the files are being typed! AND! In two ways! One method used is to create short records and the other is to create Paragraphs of data. Lets take the "short record" method first. As you type a character counter is presented in the lower RH corner of the screen. You can decide the length of each record and type away using the counter as a guide. Each time ENTER is touched the record is sent to an opened disk file. You can just keep on truckin and build an enormous length file of records, or you can quit at any time. An optional print out is provided to print any sequential file that is in disk.

The second method of Creating a Sequential file is via use of the main Daisy INPUT section, which has FULL EDIT. Here we create paragraph length bursts of data to be sent to an opened Sequential file. Again our options are many. We

can create short files of text paragraphs, or we can create an enormous length file in disk. And again, Daisy provides an efficient print out of any disk sequential file. The option is to print BY LINE (for the short records), or "PRINT to ESTABLISHED FORMAT". In the latter case a few variables are set and the incoming paragraphs of data (from disk) are sent to the Daisy Word Processor for formatted printing.

**The DAISY SYSTEMS MANAGER:** Built into the Daisy Shell program is a very efficient Managing system that manages all Peripheral devices and their functions "in coordination with MENU Selections. The management system includes; The Display Monitor, The Disk Drive, and the Printer, to provide a MENU Driven System under the complete control of the user. The management systems revolves around the FUNCTION MENU, which is "Home Base". All of the 30 menus of Daisy evolve from and go back to The Function Menu. The Function menu provides a few direct actions, but it is mostly a Selection Menu for MAJOR Functions of the software. An extension of the Function Menu is gotten by selection of "<8> OFFICE TOOLS". These two menus provide access to all functions of the software, plus the coordination with three other softwares.

The concept is to provide the user with a "Menu for Choice" for whatever function that may be needed for use. Then sub-menus are provided as needed for follow-on selections. This is opposed to the "common way" that most softwares are driven; ie, requiring a page full of key combinations to be remembered or pasted on the wall. (Hit the Caps shift 4 key while holding Symbol shift to do something, etc, ei. times 50 more!). The Daisy management system puts a menu before your eyes with subjective selections. AND, the menus are coordinated so that as major functions are elected, the next menu that appears will be related to the previously selected function.

**PRINTER CONTROL and TEXT FORMATTING** is achieved without "Junk embedded in text data", as most softwares require for commanding the printer and formatting a page. (°/°&° to do a paragraph end and line space, etc.). Instead, Daisy gives you a FORMAT MENU where you establish the format desired. If a Print style needs to be established, the STYLE MENU lets you set your print style. By use of these two menus ANY style of print that your printer is capable can be set. During the menu selection process, automatic formatting is accomplished for "Center Printed text" with whatever print style and line length is elected. The theme is "automatically formatted pages of center printed text". However, offset printing can be selected at the Format menu, and Column Printing can be elected.

**The DATA DISPLAY:** The use of the Style and Format Menus (once) eliminates the need for inserting print control symbols within the textual data. Thus a data file is always a "pure ASCII File". During Input Typing and Editing the textual data is presented with all spaces visible in an un-formatted screen. There is no correlation at all with the appearance of the text on screen with the way it will be printed. Therefore you should NOT pad in spaces for formatting or line spacing. Instead just create



groups of data with proper sentence structure and let the automatic formatting features of Daisy do the printing. Daisy will format the paragraphs with a line space between. If you have elected to have indented first lines, then Daisy does that too. So, the screen display is used for only one purpose, and that is to show you the data so that you can edit it at will. A menu of editing functions is presented at the bottom of screen so that you don't need to remember any key codes.

**DATA EDITING: INPUT TYPING:** These two important functions are intertwined. Input typing uses a machine code program for very fast keyboard response. A " " underline cursor is presented ahead of the typing input. This cursor is movable with the arrow keys for editing. When an arrow key is used the fast typing input is escaped from and the EDIT mode is entered. At this time a set of menu options are given at screen bottom. There are so many editing functions that it is difficult to list them all. First, during INPUT typing, full EDIT is there. The menu at screen bottom appears anytime you shift out of INPUT into EDIT "via the use of an arrow key." You can place the cursor anywhere within the text and make deletions, type insertions, shift into BLOCK DELETE, or BLOCK INSERT. Passages of text that are deleted with BLOCK DELETE are held, and can be re-inserted with the BLOCK INSERT function. Or, you can type a long BLOCK INSERT.

During Input Typing, the last 148 characters typed are always present. As typing continues the screen size builds until you touch ENTER. ENTER shifts the typing up on screen making room for more. UP Arrow will pull down earlier generated data to the first line for viewing or editing. When EDIT action is finished (via touching ENTER) the INPUT typing mode is re-entered. An escape from INPUT is provided. The use of CAPS and SYMBOL SHIFT together shifts out of input typing and brings the "QUICKIE MENU", which provides three more editing modes, plus other functional options. A study of the pictorials that follow will bring understanding of the variety of functions available.

The "L\$" CAPTURE: During EDITING, two functions use this CAPTURE CAPER to allow data to be "swapped about", deleted from one area and inserted at another place within the typing data—OR within any text group that has already been created and stored. A BLOCK DELETE results in TWO ACTIONS. The block of data is deleted, and the block of data is CAPTURED in the "capture buffer" which is "L\$". This data remains in L\$ until the NEXT BLOCK DELETE, upon which time the new data being deleted replaces the old. CAPTURE is a very powerful editing function that can be used in several ways. Many times you'll just want to DELETE a passage of text and discard it. Other times the passage of text may fit into another previously created data group very well. Why have to re-type a long insert?

After a data CAPTURE you can shift out of the INPUT mode, with CAPS SYM SHIFT, then elect "<2> RE-ED". Then the Quickie menu appears to allow you to select the previously stored data group for READ and EDIT. While Editing the cursor is moved to the place for "inserting the captured data". Using " " to shift into BLOCK INSERT Mode, the choice <1> at screen bottom is "<1> INSERT L\$".

ZAPPO! It is done. Another common use of the CAPTURE (Block DELETE) is when you have exceeded the data length of the H and I data cells. Remember that YOU set the limits to these 14 data groups yourself. But even though you have set a length that suits your purpose, you will often type more data than can be STORED. Say for example, you have established 1000 characters as the pre-set length of the H Files.

The situation is, you have typed the third paragraph and upon reaching the Quickie menu for STORING the data to "H-3" the menu tells you that the data length is 1500 bytes. Well, you have a dilemma. If you elect to STORE, then the last 500 characters will be lost. So, what do you do to save that 500 characters for the next paragraph? The answer is to re-enter the INPUT mode and use the cursor TO BLOCK DELETE the last 500 characters. When you do, the 500 characters will be CAPTURED in the CAPTURE BUFFER (L\$). To do that you manipulate the cursor to skip up about 17 lines to find the beginning of a sentence to begin the 500 plus character Delete. Put the cursor there and exercise a BLOCK DELETE with CAPS+EDIT. There are two steps to a Block Delete. When you move to the Start of a group to Delete and Touch ENTER, a TOOT tells you that that spot has been MARKED. Next, you move the cursor to the END of the block of text to be deleted and touch ENTER again. ZAPPO! It is done!

BUT, we will do the Block Delete differently this time because we are DELETING TO THE END. Instead of having to move the cursor down 17 lines to the end of text, just touch ENTER. That shifts into TURBO Typing mode and the cursor is put to the right of the "Last Character" of the text. So, then all we have to do is use the LEFT ARROW key to move left ONE space. Then CAPS-EDIT ZAPS the 500 characters out. This DELETES the 500+ characters and puts them in the Capture buffer "L\$". Now you can return to the Quickie Menu and STORE the DATA, which has been reduced to under your limit of 1000. The 500+ characters are safe in the Capture Buffer.

We will continue the hypothetical problem. You now can STORE the 1000 character text group by using STORE at the QUICKIE MENU. AND, you'll want to begin your next paragraph with the data that is in the Capture Buffer. To do that we select "<1> INPUT Typing" at the Quickie Menu. This gets you the Input typing screen with just the cursor in the top left of screen. The way to enter the "BLOCK INSERT MODE" is to use an ARROW KEY to escape the Turbo Typing mode. BUT, in this case you have no data to move the cursor within. It is a BLANK screen. The only way to INSERT at the beginning of the SCREEN is to use the UP ARROW ONCE. You won't see any movement, but the bottom screen menu will appear. Then the shifted " " key will give you the INSERT MENU where a touch of <1> will insert the capture buffer at the beginning of the text screen. That was a long winded dissertation to tell you how to do a very useful "TRUNCATE the TYPING DATA" and INSERT the data in the next screen.

## Important Notes on the USE OF VARIABLES

### \*\* The "varset.B6" Utility \*\*

The varset.B6 program initializes the master Variable file for the Daisy program, when called for by selection of "<9> Re-initialize Daisy". This selection is at the Office Tools Menu. Then "varset.B6" boots in and re-sets the vars file and then deletes itself. The Daisy.B6 variable file is bytes in the variable file. The variables are carefully arranged so that the ones used in routines needing the greatest speed are initialized first in the TS-2068 variable file.

Within this system vars file there are four "pseudo hex" systems for the substitution of variables for commonly used numbers. This saves a tremendous amount of FREE memory within the program. Numbers ONE through 27 are represented by the "double OH" variables, ie, zero=oo, 1=oa, 2=ob, 3=oc, through oz=25. Then since the number 27 is used quite a lot, 27=zo. The next pseudo hex table is for "thousands", where k1=1000, k2=2000, through k9=9000. The third pseudo hex table is for "hundreds", where m1=100, m2=200, through m9=900.

The fourth pseudo hex table is for "tens", beginning with 30, where t3=30, through t9=90. (10 and 20 are already represented in the "oo" table by "oj and ou"). Sometimes during the programming these pseudo hex vars will be combined, as <GO TO k6+t9> which relates to <GO to 6090>, or <GO SUB k2+m5> which relates to <GO SUB 2500>. A typical use of the "oo" vars would be <LET nn=nn+oa>, which increments the line counter nn by one (LET nn=nn+1). Other often used line numbers are represented by variables, as "fm=Line 2070". A persual through the program lines, using <PRINT vars> can give you the whole table of Line numbers that are represented by variables.

So, the programming in the lines become quite cryptic, using variables almost entirely for numbers. Any examination of the program flow will require the relationship of variables to their number representations given above. There is good reason for the encoding of the numbers to vars. At one time I counted up over 8000 bytes of FREE MEMORY that was saved within the Smart Text program, just by using variables for numbers within the programming. But all routines cannot be encrypted in such fashion. Since the use of a variable requires "look-up" time, the program operation is slowed. One example routine that must use "raw numbers" to avoid slow operation is the INPUT EDIT functions, where such use would cause an intolerable slow down.

There are also "transient" variables used throughout the program for calculations and as "switches". All of the single letter variables are temporary, and are used over and over by different routines. None of the single character vars have lasting effect, or affect from one routine to another. Incidentally, the "varset.B6" utility itself is over 2250 bytes in program length, which amount of FREE is conserved by the MERGE and DELETE concept.

Most of the MENUS have either a number or a Variable presented in the Right Bottom of the screen. These are the LINE NUMBERS of the Menu itself. EXAMPLE: the function menu has "fm" in the lower right corner. During a BREAK condition, <GO TO fm> will present the Function menu and the program will resume operation.

Some IMPORTANT variables are shown each time the Function Menu appears. These vars can be used to save "long trips" to other menus, with AN INTENTIONAL BREAK, and quick <LET>. These vars are: nn=Line Counter. In=pre-set page length. tb= TAB for left margin. ll=Line Length being formatted. qq=page center for centered print of captions. pg=current page number. To reset TAB from the existing displayed status to TAB 60: BREAK, type <LET tb=60: CONTINUE>. The Function Menu re-appears with the new TAB setting. Or, to change the page number to page 5: BREAK type <LET pg=5:CONTINUE>. The first case would allow OFFSET printing at TAB 60 (may need to change LINE LENGTH "ll" also). Such quick BREAKS and changes can speed up operation.

**\*\* LETS GET STARTED! \*\***

1. LOAD the Software. You can do this by turning ON the Computer while holding the ENTER KEY. OR, type <RANDOMIZE USR 100:LOAD"Daisy.B6">. When the software loads there will be a series of PROMPTS to set up the software for your type of Interface and printer. Your printer should be ON for this. When the few prompts have been responded to the main FUNCTION MENU will appear on screen.

2. The FUNCTION MENU: While this Imposing menu is on screen would be a good time to look at pages 8 And 9 of the manual so that you can see the logic of the menu selections. Lets create some DATA, as almost all menu electives are for managing DATA. Touch <1> Data INPUT and the Quickie Menu will appear.

3. THE QUICKIE MENU: Touch <1> TYPING. A blank screen will appear with a Cursor in the top left. This is your invitation to begin typing data. Type some data, enough to fill about a half screen. As you type and make errors, delete the error with the Shifted Delete Key. But let some errors pass on by so that we can EDIT and correct them.

4. Now lets shift into the EDIT mode by using the UP Shifted UP ARROW KEY. Move the cursor about within the text with the four Arrow keys. Try DELETING a character. You do that by placing the cursor to the RIGHT of the character and using Shifted Delete. So far so good, eh? Notice that after deleting characters the ENTER springs the cursor back to the end of the text. This puts you back into the fast INPUT typing mode for more character generation.

5. Next, move the cursor up to the beginning of one of the sentences in the middle of the text. We will DELETE that Sentence. Place the cursor under the beginning character of the sentence and use caps shift and EDIT (the one key). A foot tells you that Daisy has MARKED that character as the beginning of a BLOCK DELETE. Now use the right arrow key to move the cursor UNDER the Last Character of the sentence. When there, again use the Caps Shifted One Key. The sentence should disappear from screen (a BLOCK DELETE).

6. A BLOCK INSERT: Now place the cursor up where the sentence was before it was Deleted, using the Arrow Keys. When there, use Symbol Shift to get the British Pound (on the CLEAR KEY bottom row). A new menu will appear at screen BOTTOM. Read the menu to digest its meaning, and then touch <1>. The previously DELETED sentence will now be inserted where it was to begin with.

7. A DIFFERENT BLOCK INSERT: This time, move the cursor to some place in the text, and get the Brit Pound character again with Symbol Shift and the pound key. Again the new menu will appear at screen bottom. This time touch <2>. The Quotes for string input will appear. Type some characters and then touch ENTER. This block of characters will be INSERTED from where the cursor was positioned. Now, for practice, BLOCK DELETE the last bunch of characters that you inserted.

8. ESCAPE FROM INPUT: Use Caps Shift and Symbol Shift together. The Quickie Menu will appear. Now that you have some data to work with,

the amount of characters will be reported in the Information section of the menu. Lets STORE THE DATA. Touch <3> STORE. A prompt will appear asking you to decide "The maximum Data Length for the paragraphs that you want to generate and STORE. A good figure to INPUT would be about 800. INPUT 800.

9. Now notice the new data presented at bottom left of the Quickie menu. "H=2" means that the NEXT data group that you STORE will go into the data base "H-2". Also notice that the "Cell Limit" is reported as 800, and the Typing Buffer is at Zero capacity. These Information products guide you for subsequent data management.

10. Next, touch <1> to get back into the INPUT TYPING Mode. Again the blank Screen will appear. This time use the UP ARROW key ONCE. Then use Symbol Shift and the British Pound. The BLOCK INSERT Menu should come to bottom of screen. Touch <1>. The text data that you previously DELETED from the STORED Paragraph will be inserted. Type some more characters and then Move the cursor up to the to the Beginning of the next sentence. Again, use the Brit POUND. Touch <1> again. Again, the SAME data will be inserted, which illustrates that the "CAPTURE BUFFER" is preserved until "something else" is put into it.

11. Next, shift OUT of the INPUT Mode. Remember HOW? (Caps and Symbol Shift). This time when the Quickie Menu appears, touch <9> PASTE UP. Now type some characters and touch ENTER. Now you have inserted different character data into the CAPTURE BUFFER, and it will stay there until you make another BLOCK DELETE.

12. Touch <1> again. The previously typed data will appear on screen. Move the cursor up to the beginning of a sentence and get the Brit Pound again. Touch <1>, which will demonstrate the Block Insert to Paste-Up text.

13. Return to the Quickie Menu and STORE the paragraph. Then touch <2>. WOW! what a busy new menu! Study it a bit. This is the EDIT MENU for data that has previously been STORED. ENTER <1> to get the first paragraph that was Stored. Now run the cursor up to the beginning of a sentence. Get the Brit Pound for an insert. Touch <1> and watch the data that is in the Capture buffer insert itself in the text.

14. Now touch ENTER, and watch the cursor spring to the END of the TEXT. Actually, it goes to END of 800 characters, and the last part is nothing but spaces. Use the UP arrow to position the cursor to two spaces beyond the end of the last sentence. Now use the Brit Pound for a Block Insert. Type a insert and touch ENTER. This just demonstrates how you can recall previously STORED data, use the Block Insert to add on more text-- of course "within the 800 character limit" that you input for data length. Now to get back to the Quickie Menu, use Cap and Symbol Shift.

15. Now to do something else. Since you have TWO data groups STORED, lets recall them and EDIT BOTH in sequence. At the Quickie Menu, touch <6>.

But the menu at bottom is different. This time you are in a "READ with EDIT CHOICE". Note that to EDIT, you touch the "E" key. But lets DONT. Just touch ENTER. Now you have the second group of text on screen. Touch "E" for Edit. Now you have the same type of Full EDIT functions previously found with the other menu. To get OUT of EDIT, use Caps and Symbol Shift. This brings the SAME text back to READ Again, and EDIT MORE if needed. Otherwise ENTER will get the third group of STORED data.

16. Now touch ENTER. Since you haven't yet STORED a third group of text, you will be viewing and EMPTY text cell. The capability is to View and EDIT 7 cells of data in each of the two data bases "H and I". Touch ENTER again and note that the information at bottom identifies the data as "H-4". A little trick to get back to the Menu is to Jab Break. Do that. Now you are back at the "read edit" menu where you could select <6> to view and edit the 7 groups of the "I data base". Instead, knowing that there is no data there, Enter to select the Quickie Menu.

17. Now back at the Quickie menu, select <1> Input again and type some characters. Use Caps Sym Shift to get back to the Quickie menu. This time touch <5>. You have just cleared the typing buffer.

18. Now to complete the sojourn through all Quickie Menu items, touch <7>. Now we will replace the word "The" with "Licorice" (just for a humorous kick). Enter the word "The" as the word to replace, and "replace with licorice". Have a cup of coffee or something. This is a slow routine, especially if you have all 14 data cells filled with long stored groups.

19. Item <0> is for EDITING a Sequential File that has been converted to a character Array. But we are not ready for that yet. Item "<:>" (colon key) lets you clear the data base and start fresh.

20. Now that we have some data to print, lets do it. Select <4> Help, which gets the Function Menu. Next select <8> Office Tools, and then <4> Print Menu. Here you will have choices of printing functions. Select <2> Letters. Follow the prompts and Input 2 copies. This will be the acid test of whether you need to work on "Matching your Printer" to the software. If the letter prints out "center printed" with a "center printed letterhead" the HOORAY! If not the B-oweee. You have some work to do on your printer matching. There is a section of the manual about that subject.

You have become familiar with the management of "Just one" of the Daisy Data bases. This is a good start and you can use this for creating and printing Text Data files. The next step is to check out all of the functions of the FUNCTION MENU.

#### THE FUNCTION MENU

As you have seen, after a function has finished, the Function Menu RE-Appears. If you had trouble with the previous step, then a "report

code" would come at the screen bottom, which would indicate that you had either done something wrong, or your printer is not matched to the software. Any report code can be recovered from by typing <GOTO fm ENTER>.

Now at the Function Menu, lets take a Safari through the "Menu Wilderness". It will be educational. First touch <4> to see the Data Status Menu. This menu presents the status of all text data bases and the Mail File. Study in for a while. It is very informative, and is useful to help manage the data bases. Simple string data is also reported, but not managed by the software. To put data into simple string you need to BREAK and do a LET statement. When you have finished studying the Status Report just touch ENTER to return to the Function Menu. Now lets do that again with a different variation. Select <4> Status Report, and when the report comes JAB BREAK. This is the quickest way of "cleaning out all data" and starting afresh with a clean program.

The selection <1> Yes will do that by doing a CLEAR, and then a re-load of the clean variable file "01697.C2". But lets dont do that right now because I want to use some of the data that is stored in your program. Instead ENTER <2> to return to the Function Menu. At the Function Menu, touch <5> for the Utility menu. The Utility Menu comes on with nine options. Study the menu for a while. Notice that item <5> offers the option of returning to the Function Menu. Most of the Daisy menus provide this option so that you can easily abort and get back to first base. Select <2> DELETE MENU. Now we have a menu that does many things. The first five selections deal with the data bases.

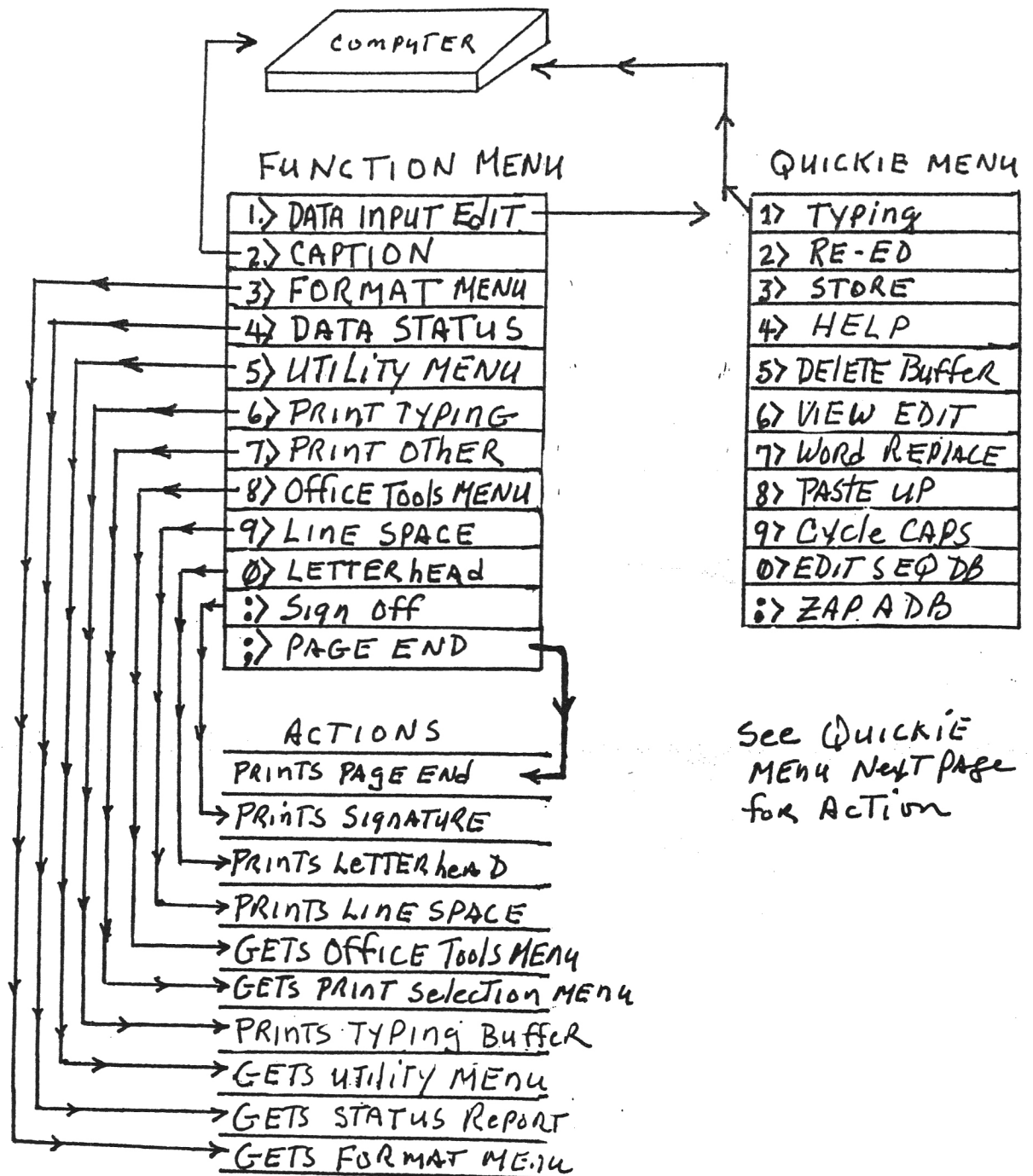
Each of the data bases may be CLEARED, or ALL may be CLEARED. The current status of FREE Memory is presented at top right. Item <5> is the same type of function that we discussed with the Status Report. A selection of <5> CLEARS all data and re-loads the clean vars file. Item <6> Turns "Turbo Off" and deletes all auxillary programs that may be "attached" to the main Daisy program. After that selection, the individual utilities MERGE to Daisy when called for by a Menu operation. This makes the software have short delays while the MERGE takes place, but provides a much larger FREE MEMORY capacity for large data bases.

Now at the DELETE MENU select <8> to return to the Function menu. At the function Menu, again select <5> Utility Menu. This time at the Utility Menu, elect <3> Load Menu. Here we have the ability to quickly LOAD in data files from disk. If you selected <1> H Array, the data loaded in would REPLACE the two data groups that you have in your program now. If you wanted to keep this data in program and LOAD a "I data base" then <2> would do it. Then you could print the two EXISTING data groups and the new group that may have 7 more data groups. The two groups would print in series (As 9 text groups).

Continue Investigating the SAVE Menu and others, working you way through all software menus. The Function Menu operates as a "rotary" to access all other menus, and is returned to as functions are completed. This cursory start should allow you to do simple data base input, Edit, and printing.

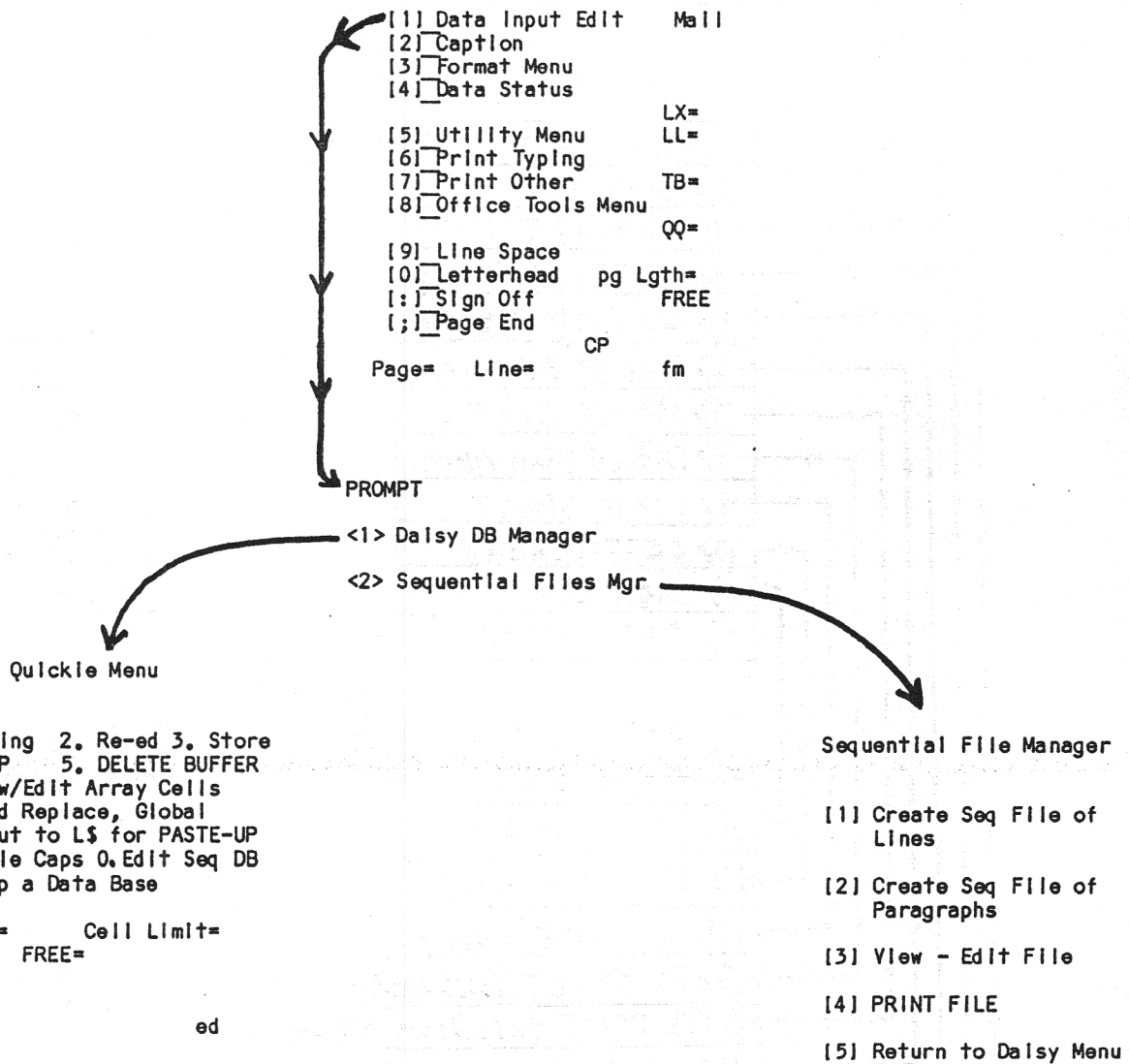


# THE FUNCTION MENU



★ See Other MENUS  
Succeeding PAGES

# FUNCTION MENU



The Function Menu is "HOME BASE" for all of Daisy's Functions. You can always reach the Function menu from other Menus, OR, in a subscript BREAK condition, via <GO TO FM ENTER>.

The Illustration above shows the menu routing when "<1> Data Input-EDIT" is elected at the Function Menu. Daisy has several "Data Base Managers". The Quickie Menu (Left Illustration) is the main data base manager menu for the two Text Data Bases "H and I". The Sequential Files Data base is managed with the Menu at right.

The Illustrations that follow will deal with the coordination between the Function Menu and the QUICKIE MENU for Data Base Management.

\*\*\* Program "daz.B6" LLIST \*\*\*

7545 Bytes

PAGE #  
OF MERGE  
PROGRAM

NOTES

The DAISY SHELL PROGRAM "Daisy.B6"

This Main LISTING is given mostly for reference by those who want to delve into the programming. It is not necessary for OPERATIONAL purposes to know the details of program line logic. But such detail is given as the "MERGE PROGRAMS" are discussed. This listing is the "DAISY SHELL MANAGER" program, which is 3 1/2 pages of condensed print. I will write in the side board margins to bring more information, mostly about CONCEPTS and operation.

AUTO  
START

```

498 CLS : PRINT AT VAL "10",VAL "2";"Please Stand
By. . .": RETURN
500 RANDOMIZE USR 100: LOAD "dpr.C1"SCREEN$: PAU
SE 0: LET z=CODE INKEY$-48
502 IF z<1 OR z>3 THEN GO TO VAL "502"
504 IF z=1 THEN RANDOMIZE USR 100: MERGE "varsv. - 28
B6": LET alt=508: GO TO 9902
506 IF z=3 THEN GO TO fm
508 RANDOMIZE USR 100: LOAD "dprtr.B6" 4 29
600 RANDOMIZE USR m1: MERGE "cat.B6": GO TO VAL " - 37
9822"
700 RANDOMIZE USR 100: MERGE "scopy.B6": GO TO VA - 30
L "7004"
840 GO SUB VAL "495": RANDOMIZE USR m1: MERGE "tb - 25
.B6"
900 ON ERR RESET : ON ERR GO TO VAL "902": DELE
TE VAL "842",VAL "877": STOP
902 ON ERR RESET : GO TO alt
1000 RANDOMIZE USR m1: LOAD "scopy.B6" - 30
1895 CLS : GO SUB VAL "495": RANDOMIZE USR m1: MER
GE "mail.B6": GO TO k2 - 32
2057 DELETE VAL "1896",VAL "2055": GO TO fm
2060 INK 7: LET turbo=oa: POKE VAL "23730",VAL "87
": POKE VAL "23731",VAL "255": RANDOMIZE USR m1: M
ERGE "init.B6" - 14
2069 DELETE VAL "2061",VAL "2068"
2070 ON ERR RESET : RANDOMIZE USR m1: LOAD "fm.C1
"SCREEN$: LET seq=oo: LET alt=fm: LET gg=oo: LET
nt=nz: LET mzk=oo
2071 PRINT AT op,VAL "28"; INK oe;ln;AT og,zo;("ON
" AND turbo>oo)+("OFF" AND turbo<oa);AT or,oz; FR
EE ;AT ot,ot;cp;AT ob,VAL "26";("pica" AND ps=oa)+
("Elite" AND ps=oc)+("Cond" AND ps=od)+("PICA" AND
ps=ob);AT oh,VAL "27";lx;" ";AT ol,VAL "28";ll;AT
ok,VAL "28";tb;AT om,VAL "28";qq;AT ot,og;pg;AT o
t,op;nn;AT oe,VAL "26"; INVERSE oa;("Yes" AND LEN
o$(oa,oa)>oa)+("No " AND LEN o$(oa,oa)<ob); INVERS
E oo: GO SUB sq: PRINT AT oo,VAL "27"; INK oe; FLA
SH oa;"a"; INK og: GO SUB ll: IF z<oo OR z>ok THEN
GO TO VAL "2071"
2072 IF z=ok THEN GO SUB pe: GO TO fm
2073 IF z=ol THEN BEEP ob/oj,ov: GO SUB sp: GO TO
VAL "2071"
2074 IF z=of THEN LET w$=u$: GO SUB te: GO TO fm
2075 GO TO (z=oa)*VAL "3416"+(z=ob)*VAL "2094"+(z=
oc)*VAL "2078"+(z=od)*VAL "3579"+(z=oe)*VAL "4099"
+(z=oh)*VAL "4050"+(z=og)*VAL "3700"+(z=oo)*VAL "2
592"+(z=oj)*VAL "4090"
2076 LPRINT !: LET nn=nn+oa: IF nn>=ln THEN GO SU
B pe
2077 RETURN
2078 GO SUB VAL "498": RANDOMIZE USR m1: MERGE "fo - 22
mat.B6"
2092 DELETE VAL "2080",VAL "2091": GO TO fm
2093 DELETE VAL "2080",VAL "2091": GO TO k8
2094 CLS : PRINT AT oj,od;"Type the Cap"TAB od;"
Ln lgth limit ls: ";lx: INPUT m$: CLS
2095 LPRINT TAB qq-LEN m$/ob;m$: LET nn=nn+oa
2096 IF gg=oa THEN RETURN
2097 GO TO fm
2098 LET tel=VAL "3742": IF turbo<>ob THEN CLS :
GO SUB VAL "498": RANDOMIZE USR m1: MERGE "repp.B6 - 19
"
2180 CLS : LET rpr=oa: PRINT AT oo,oj;"Event # ";l
: LET zp=zx: IF turbo<>ob THEN GO SUB VAL "498": - 20
RANDOMIZE USR m1: MERGE "usrpgm.B6": RANDOMIZE USR
m1: MERGE "wdpro.B6" - 21
2181 CLS : PRINT AT oj,oo;"Copy # ";l: REM LET m$
="HEADER": GO SUB VAL "2095": GO SUB sp
2410 ON ERR GO TO VAL "2411": IF turbo<>ob THEN
DELETE VAL "2100",VAL "2174": DELETE VAL "2182",VA
L "2403": DELETE VAL "3742",VAL "3812": STOP

```

MENUS AND TEXT  
ARE NOT Related.

MENUS SUPPORT LISTING

AUTO START Line IS 2060

SCREENS Load  
From Disk

| FUNCTION MENU         |          |
|-----------------------|----------|
| 11) Data Input Edit   | Mail     |
| 12) Caption           |          |
| 13) Format Menu       |          |
| 14) Data Status       |          |
| 15) Utility Menu      | LX=      |
| 16) Print Typing      | LL=      |
| 17) Print Other       | TB=      |
| 18) Office Tools Menu | QQ=      |
| 19) Line Space        |          |
| 10) Letterhead        | pg Lgth= |
| 11) Sign Off          | FREE     |
| 12) Page End          |          |
|                       | CP       |
| Page=                 | Line= fm |

<GO TO FM ENTER>

"The Cure To What Ails You"  
BETT'RN Pepta-Bismo  
For Subscript ERRORS.

**\*\* General Information \*\***

The DAISY SHELL is just that. It contains the "Software Manager" and some utilities that stay aboard. FREE MEMORY upon loading is about 21K. The general concept is to break the large software into about 25 functional elements, all stored in disk. As the elements are needed for operation they MERGE IN to the main SHELL program. When the functions that are inclosed in the MERGE elements operate, the trip to the Function Menu DELETES the "Merge.B6" program, thus keeping FREE MEMORY at Maximum.

The TURBO MODE: Early on it was discovered that programs that are longer than about 2K in program length take about 12 seconds to complete a MERGE-IN. 12 seconds does not sound like a long time, but when doing data management it seems like an eternity. So, a TURBO Mode was devised, where the Data Base INPUT and EDIT functions are "stuck to" the Shell program. Also, since a close cousin to the Data base functions is PRINTING, the printing functions are "attached" when in the TURBO 2 mode.

When first LOADED, via <LKDOS AUTO LOAD> or via <RANDOMIZE USR 100:LOAD "Daisy.B6"> the mode is TURBO 1. We actually use a "variable switch" to denote TURBO STATUS. The switch is variable "turbo". "turbo=0" when in the OFF mode and "turbo=1" for DATA BASE MANAGEMENT, TURBO 2 for Printing. FREE MEMORY is reduced to about 21K. But, you don't have to wait for the most often used functions to MERGE in during switching back and forth between the Function Menu, the INPUT section, the EDIT section, and the PRINTING sections.

You, the operator, have control of the TURBO MODE from two Menus. To turn Turbo OFF, visit the OFFICE Tools Menu. The route is from the Function Menu Item <8>, then ITEM <1> at OFFICE Tools. The result of selecting TURBO OFF is that these

```

2411 ON ERR RESET : ON ERR GO TO VAL "2412": DEL
ETE VAL "4055",VAL "4088": STOP
2412 ON ERR RESET : GO TO fm
2550 PAUSE oo: LET t$=INKEY$: IF t$="" THEN GO TO
11
2552 LET z=VAL "-48"+CODE t$: RETURN
2554 PAUSE oo: LET t$=INKEY$: IF t$="" THEN GO TO
1k
2556 IF t$<>"y" AND t$<>"Y" AND t$<>"n" AND t$<>"N
" THEN GO TO 1k
2558 IF t$="y" OR t$="Y" THEN LET z=oa
2559 IF t$="n" OR t$="N" THEN LET z=ob
2560 RETURN
2570 CLS : LET m$="SM SAVE": LET w$="SM": GO SUB
k5+m2: INPUT "Input File Nbr: ";c: LET w$="SM"+STR
$ c+"B1": RANDOMIZE USR m1: SAVE w$ LINE VAL "206
0": RANDOMIZE USR m1: SAVE "cat,C1"CODE VAL "24311
",VAL "265": GO TO VAL "5120"
2574 BEEP oa/ob,od: BEEP oa/ob,oh: BEEP oa/ob,oj:
BEEP oa/ob,od: GO SUB sq: RETURN
2576 INK xa: PLOT oo,oo: DRAW oo,VAL "175": DRAW V
AL "255",oo: DRAW oo,-VAL "175": DRAW -VAL "255",o
o: LET xa=xa+oa: IF xa>og THEN LET xa=oa
2577 INK xa: PLOT oh,oh: DRAW oo,VAL "159": DRAW V
AL "239",oo: DRAW oo,-VAL "159": DRAW -VAL "239",o
o: LET xa=xa+oa: IF xa>og THEN LET xa=oa
2580 INK og: BEEP oa/oj,VAL "40": RETURN
2590: RANDOMIZE USR m1: MERGE "lh.B6": RETURN
2592 RANDOMIZE USR m1: MERGE "lh.B6": GO TO VAL "4
055"
2594 DELETE VAL "4055",VAL "4088": GO TO fm
2598 RANDOMIZE USR 100: LOAD "vcalc.B6"
2600 CLS : IF turbo<>oa THEN LET turbo=oa: LET al
t=VAL "9668": GO SUB VAL "395": RANDOMIZE USR m1:
MERGE "tb.B6": GO TO VAL "850"
2603 GO TO VAL "9668"
2604 IF turbo<>oa THEN DELETE VAL "9668",VAL "973
7": ON ERR GO TO VAL "2605": DELETE VAL "3742",VA
L "3812": STOP
2605 ON ERR RESET : GO TO alt
3579 RANDOMIZE USR m1: MERGE "st.B6": GO TO VAL "3
580"
3588 ON ERR RESET : DELETE VAL "3580",VAL "3584":
GO TO k6+t5
3589 DELETE VAL "3580",VAL "3584": GO TO fm
3700 RANDOMIZE USR 100: MERGE "pt.B6"
3719 DELETE VAL "3701",VAL "3718"
3741 IF turbo<>ob THEN RANDOMIZE USR m1: MERGE "w
dpro.B6"
3814 IF turbo<>ob THEN DELETE VAL "3742",VAL "381
0"
3816 RETURN
3900 LET alt=VAL "3902": GO SUB VAL "495": LET tur
bo=oo: RANDOMIZE USR m1: MERGE "tb.B6": GO TO VAL
"850"
3902 CLEAR
3904 RANDOMIZE USR VAL "100": MERGE "btls.B6": GO
TO VAL "3912"
4041 DELETE VAL "3912",VAL "4040": RANDOMIZE USR V
AL "100": MERGE "cho.B6": GO TO VAL "3912"
4042 DELETE VAL "3912",VAL "4040": RANDOMIZE USR V
AL "100": MERGE "mzrt.B6": GO TO VAL "3912"
4043 DELETE VAL "3912",VAL "4040": GO TO VAL "3904
"
4044 ON ERR RESET : DELETE VAL "3912",VAL "4040":
RANDOMIZE USR VAL "100": MERGE "varset.B6": GO TO
VAL "9994"
4046 DELETE VAL "3912",VAL "4040": GO TO VAL "3904
"
4050 ON ERR RESET : RANDOMIZE USR m1: LOAD "ot.C1
"SCREEN$: LET zy=oo: LET zx=oo: LET zm=oo: LET mz
=oo: LET az=oo: LET no=oo: LET nx=oo: LET z5=oo: L
ET df=oo: LET scp=oo: LET mm=oo: PRINT AT od,ow; I
NVERSE oa;"turbo=";turbo; INVERSE oo

```

See Office  
Tools Menu  
Next Page

**DATA STATUS**

AS=0 par in H\$=  
BS=0 par in I\$=  
CS=0  
DS=0 FREE=  
ES=0  
FS=0 Ent Whn Rdy  
GS=0

LS=0  
QS=0 NO  
RS=0 MAIL  
JS=0 DATA  
SS=0  
US=0

Use K\$, P\$, V\$ for spare  
Data ARRAYS 3579

```

4051 GO SUB sq: GO SUB ll: IF z<ov AND z>oi THEN
LET z=z-og
4052 GO TO (z>ok)*VAL "4051"+(z=oa)*VAL "840"+(z=ob)*VAL "500"+(z=oc)*VAL "6700"+(z=od)*VAL "2098"+(z=oe)*VAL "1895"+(z=of)*VAL "2598"+(z=og)*fm+(z=oh)*VAL "5400"+(z=oi)*VAL "9994"+(z=oj)*VAL "2600"+(z=ok)*VAL "3900"+(z=oo)*VAL "8120"
4090 LET qq=(qq AND cp=oo)+(VAL "34" AND cp=oa)+(VAL "98" AND cp>oa): LPRINT : LPRINT : LPRINT : LPRINT TAB qq;n$(ok): LPRINT : LPRINT : LPRINT : LPRINT TAB qq;n$(om): LET nn=nn+oh: IF scp=oa THEN GO SUB k8+m2
4091 IF rpr=oo THEN GO SUB pe
4092 IF gg=oa THEN RETURN
4094 IF rpr=oa THEN RETURN
4098 GO TO fm
4099 RANDOMIZE USR 100: MERGE "dc.B6": GO TO VAL "5160"
5212 DELETE VAL "5100",VAL "5210": GO TO (z<oe)*VAL "3579"+(z=oe)*fm+(z=of)*VAL "600"+(z=og)*k1+(z>og)*VAL "3900"
5213 DELETE VAL "5100",VAL "5210": GO TO (z=oe)*VAL "9900"+(z=of)*fm
5214 DELETE VAL "5100",VAL "5210": GO TO (z=oe)*VAL "6050"+(z=of)*VAL "840"+(z=og)*VAL "3579"+(z=oh)*fm
5215 DELETE VAL "5100",VAL "5210": RANDOMIZE USR m1: CAT ".A$": PRINT #oo;"At STOP, ENTER Type LOAD Command": PAUSE k1: STOP
5216 DELETE VAL "5100",VAL "5210": GO TO ait
5217 DELETE VAL "5100",VAL "515210": RANDOMIZE USR m1: GO TO VAL "9900"
5218 DELETE VAL "5100",VAL "5210": GO TO fm
5300 CLEAR VAL "65367": DELETE VAL "5226",VAL "5290": RANDOMIZE USR VAL "100": POKE VAL "16090",VAL "200": RANDOMIZE USR VAL "100": POKE VAL "16094",VAL "0": LPRINT : RANDOMIZE USR VAL "100": LOAD "cc1r.C1"CODE: DIM k$(8): DIM v$(VAL "1630"): LET k$="01638.C2": RANDOMIZE USR VAL "100": MERGE "varsv.B6": GO TO VAL "9990"
5400 RANDOMIZE USR m1: MERGE "pg.B6"
5410 DELETE VAL "5401",VAL "5408": GO TO fm
5890 RANDOMIZE USR m1: MERGE "pe.B6"
5914 DELETE VAL "5900",VAL "5913": RETURN
6050 ON ERR RESET : CLS : PRINT AT VAL "10",VAL "0";"Complete DATA DELETE and LOAD of Variable File was Ordered? CONFIRM""<1> Yes -or- <2> No": PAUSE NOT PI: LET z=CODE INKEY$-VAL "48": IF z<oa OR z>ob THEN GO TO VAL "6050"
6052 IF z=oa THEN CLEAR : RANDOMIZE USR VAL "100": MERGE "varsv.B6": POKE VAL "24374",VAL "22": POKE VAL "24375",VAL "8": LET k$="01697.C2": DIM v$(VAL "1685"): GO TO VAL "9982"
6054 GO TO fm
6700 CLS : RANDOMIZE USR m1: LOAD "tbo.C1"SCREEN$

6701 GO SUB sq: GO SUB ll: IF z<oa OR z>oc THEN GO TO VAL "6701"
6702 IF z=oc THEN GO TO fm
6704 IF z=ob THEN GO TO VAL "6712"
6705 CLS : PRINT AT oj,ob;"Please Identify the number of TAB ob;"This PROGRAM DISK": INPUT "Input Disk Number";d: POKE VAL "24373",d
6707 RANDOMIZE USR m1: MERGE "varsv.B6"
6709 POKE VAL "24375",VAL "26": POKE VAL "24374",VAL "51": GO TO VAL "9902"
6710 POKE VAL "24375",VAL "26": POKE VAL "24374",VAL "51": PRINT AT VAL "18",NOT PI; FLASH NOT NOT PI;z$: FLASH NOT PI;" Is the file name SAVED""Make Note, then ENTER": LET d=PEEK VAL "24373": RANDOMIZE USR VAL "100": GO TO d
6711 CLS : PRINT AT VAL "18",NOT PI; FLASH NOT NOT PI;z$: FLASH NOT PI;" Is the file name SAVED""Make Note, then ENTER": LET d=PEEK VAL "24373": RANDOMIZE USR VAL "100": GO TO d

```

"Merge.B6" programs are DELETED. Then each function that requires one of the ".B6" programs will result in a 3-12 second wait for the merge to be accomplished.

#### Select Data to Print

|            |            |
|------------|------------|
| <1> H\$(1) | <8> I\$(1) |
| <2> H\$(2) | <9> I\$(2) |
| <3> H\$(3) | 10> I\$(3) |
| <4> H\$(4) | 11> I\$(4) |
| <5> H\$(5) | 12> I\$(5) |
| <6> H\$(6) | 13> I\$(6) |
| <7> H\$(7) | 14> I\$(7) |

|               |         |
|---------------|---------|
| 15> Typing    | 18> C\$ |
| 16> A\$       | 19> D\$ |
| 17> B\$       | 20> E\$ |
| 21> Abort     |         |
| Enter Choice: | prt     |

#### OFFICE TOOLS

[1] Turbo Change turbo=1  
[2] Print Code ASCII File  
[3] Load Dbase-1  
[4] Repeat Print Menu

[5] Load Mail Merge  
[6] Load View Calc  
[7] To Function Menu  
[8] Page Management

[9] Re-initialize Daisy  
[10] Create/Prt Outline DB  
C [A] LKDOS Seq DB Manager  
A [B] LKDOS Desk Top  
P (Temp Substitute)  
S

4050

TURBO ON is selected by item <1> at the Office Tools Menu. When selected, the programs, "Inpt.B6" merges. This is the fast INPUT typing program and the EDIT functions.

"Turbo 2" mode was added for printing speed. When in Turbo 2, the "REPP.B6", "USRPM.B6" and "WDPRO.B6" programs are attached. And the DATA Base programs are Deleted.

The advantages of using TURBO 1 is its speed of Data base Management and Printing selections. Each function does not operate faster, but the transfer between functions are streamlined. The practical data capacity for TURBO ON is about 2/K.

The advantages of operating in TURBO OFF mode is a much greater Data capacity, about 30K, and the ability to operate ALL of the ".B6" annexes without fear of compromising FREE MEMORY. You can have quite a large



```

6712>RANDOMIZE USR VAL "100":LOAD "dbase1.B6"
7092 DELETE VAL "7004",VAL "7090": GO TO fm
8000 CLS : RANDOMIZE USR m1: MERGE "stymn.B6" — 23
8114 ON ERR GO TO VAL "8115": DELETE VAL "8001",V
AL "8110": DELETE VAL "9202",VAL "9208": DELETE VA
L "9500",VAL "9520": STOP
8115 ON ERR RESET : GO TO VAL "2078"
8120 INPUT "Install Pgm Disk and ENTER #":dd: RAND
OMIZE USR m1: GO TO dd: RANDOMIZE USR m1: MERGE "o
tln.B6" — 24
8176 DELETE VAL "8121",VAL "8168": GO TO fm
8200 LPRINT '': LET nn=nn+1: LET w$=u$: GO SUB +
e: RETURN
8990 RANDOMIZE USR m1: MERGE "df.B6" — 27
9024 DELETE k9,VAL "9022": GO TO VAL "2100"
9025 DELETE k9,VAL "9022": GO TO VAL "4050"
9380 RANDOMIZE USR m1: MERGE "word.B6" — 27
9492 DELETE VAL "9400",VAL "9488": GO TO (turbo>oo
)*ed+(turbo<oa)*VAL "3416"
9866 DELETE VAL "9822",VAL "9864": GO TO fm
9900 RANDOMIZE USR m1: MERGE "varsv.B6" — 28
9992 DELETE VAL "9902",VAL "9991": GO TO PEEK VAL
"24374"+VAL "256"*PEEK VAL "24375"
9994 POKE VAL "24375",VAL "8": POKE VAL "24374",VA
L "22": RANDOMIZE USR VAL "100": MERGE "varset.B6" — 31
: GO TO VAL "9992"
9995 POKE VAL "24375",VAL "8": POKE VAL "24374",VA
L "22": RANDOMIZE USR VAL "100": MERGE "varsv.B6": — 28
GO TO VAL "9992"
9996 POKE VAL "24375",VAL "8": POKE VAL "24374",VA
L "22": RANDOMIZE USR VAL "100": MERGE "varsv.B6": — 28
GO TO VAL "9992"
9997 GO TO VAL "3010"

```

END of DAISY SHELL MANAGER

Some Valuable Hidden Functions

Normally the ".B6" annex programs are NOT ON BOARD the Daisy Shell program. The .B6 programs are automatically MERGED AS NEEDED. Sometimes it may be useful to BREAK IN "after" a merge has taken place to make special one-time edits of the merged ".B6" program. But such editing actions will have only temporary effect. When the .B6 program "leaves the shell", the next MERGE of the same program will not have your modifications. To make permanent changes, the .B6 program must be loaded and changed, then SAVED OVER the original .B6 program.

\*\*\* SOME KEY <GO TO's>: For use when the program has been interrupted: <GO TO FM> gets the Function Menu. <GO TO ed> gets the EDIT Menu. <GO TO 4050> gets the Office Tools Menu. <GO TO 2060> Re-initializes the Daisy program without data loss. <GO TO 6050> Re-loads the clean vars file.

NICE TO KNOW, and sometimes useful: <GO TO pe> executes a page end. <LET M\$="data":GO TO te> If TURBO is OFF this will MERGE the word processor and print M\$ in the current format. <LET M\$="data":GO SUB tel, if turbo is ON will print M\$ data. <LET M\$="data":GO TO 2096> will print a centered header. <LET M\$="data":GO SUB 2095>, if in a program line in the User Programming Group, will print a centered caption ahead of the data to be printed. Line number 2181 is a convenient place to schedule the "leading header" for the whole text group.

data base in memory and still operate all of the ".B6" annex functions when TURBO is OFF. So, you should always be cognizant of the TURBO MODE that you are operating with, and TURN TURBO OFF when not involved in data base input, edit, and printing. It is very easy to reverse the TURBO Mode at menu.

#### LOAD MENU

1. Load In H\$ Array file
2. Load In I\$ Array file
3. Load In Mail List
4. Load In a Var file
5. Other type of LOAD
6. To Function Menu

5100

Current Page is pg 1

Line to pg end=52

Select

- <1> Page end
- <2> Adjust line to pg end
- <3> Re-num Page
- <4> To fun menu

pm

#### Utility Menu

- 111 SAVE Menu
- 121 Delete Menu
- 131 LOAD Menu
- 141 Status Report
- 151 Function Menu
- 161 Copy Catalog
- 171 Screen Copy
- 181 Number Tables
- 191 Music Interlude

5160

#### DELETE MENU

- 111 Mail Data
- 121 H\$ Data Base
- 131 I\$ Data Base
- 141 Spare Stores
- 151 All Data
- 161 All Merge .B6 Programs
- 171 To Status Report
- 181 To Function Menu 5168

#### SAVE MENU

- <1> DAISY Prog and Data
- <2> Mail Data
- <3> H\$ Data
- <4> I\$ Data
- <5> Disk Save All Data
- <6> To Function Menu

5164

# "Init,B6" LKDOS Initialization

```

2061 BORDER oo: PAPER oo: INK og
: CLS : RANDOMIZE USR m1: LOAD "
cpr.C1"SCREEN$: RANDOMIZE USR m
1: LOAD "cat.C1"CODE : PRINT #oo
;"PRESS A KEY...": GO SUB VAL "2
574": PAUSE oo: CLS : PRINT AT o
j,od;"Touch:""TAB od;"<1> For T
S-2040 Printer""TAB od;"<2> For
Dot Matrix""TAB od;"<3> For Da
isy Wheel": GO SUB sq: GO SUB il
: LET pr=z-oo: IF z<oa OR z>oc T
HEN GO TO VAL "2061"
2062 IF pr=oo THEN RANDOMIZE US
R m1: OPEN #og,"LP": LET il=VAL
"32": LET qq=VAL "16": LET ix=il
: LET tb=oo: RANDOMIZE USR m1: P
OKE VAL "16092",oo: GO TO VAL "2
069"
2063 RANDOMIZE USR m1: LOAD "if.
C1"SCREEN$: GO SUB il: GO SUB s
q: LET a=z-oo: PRINT #oo;"Printe
r Need Line Feed? <y> <n>": GO S
UB ik: GO SUB sq: LET b=z-oo: RA
NDOMIZE USR m1: OPEN #3,"lp": RA
NDOMIZE USR m1: OPEN #4,"dd": RA
NDOMIZE USR m1: POKE VAL "16096"
,a: RANDOMIZE USR m1: POKE VAL "
16090",VAL "136": RANDOMIZE USR
m1: POKE VAL "16092",b"oj: RANDO
MIZE USR m1: POKE VAL "16094",oo
: LPRINT
2068 CLS : PRINT AT oj,oh;"Rt Ma
rgin Justify?""TAB oh;"<Y> Yes
or <N> No": GO SUB sq: GO SUB ik
: LET jy=z: PRINT AT oj,oh;"Firs
t Line Indent?""TAB oh;"<Y> Yes
or <N> No": GO SUB sq: GO SUB i
k: LET nz=-z+ob

```

## \*\* The FIRST ON Initialization \*\*

When Daisy is first loaded these prompts allow you to select Your type of Parallel Centronic Interface, and then initializes the LKDOS system for that. The software assumes a DUMB printer, and sets up LKDOS for the MAXIMUM Printer Line. All printers may be set up for either a LINE FEED as each carriage return occurs, or NO LINE FEED. The user group is split about 50/50 as to their Line Feed Choice of printer setting. This prompt allows the use of the program with whatever printer setting is the user's standard operating mode. One can operate Daisy with MANUAL printer settings by SELECTING DAISY WHEEL PRINTER at the prompts that follow during Initialization. This disables the "Dot Matrix Printer Control Section" that will be described later in the manual, and allows printing to be accomplished using MANUAL printer settings for the PRINT STYLE. The Dot Matrix Printer Control sections provides AUTOMATIC PRINTER STYLE CHANGES via MENU ELECTIVES.

The programming for these style changes are based upon EPSON COMPATABLE PRINTERS. If your printer is different, and does not respond correctly to Automatic Style Changes selected at menu, then while waiting for the opportune time to revise the programming, just select DAISY WHEEL during the Initialization prompts. Then you can use the front panel switches of the printer to select the style of print desired. When operating in the Daisy Wheel Mode, the STYLE MENU is still

\*\*\*\*\*

\*\*\*\*

## DAISY

The TS-2068 Software System

For The

Larken Disk System

Copyright 1989 by:  
Gulf Micro Electronics  
1317 Stratford Ave.  
Panama City, FL 32404  
(rights reserved)

Select the INTERFACE  
That you use:

<1> Ollger or AERCO

<2> Tasman CPI

<3> A & J CPI

2067

used for the mathematics calculations for MATCHING LINE. Read the section that follow about FORMATTING and the STYLE MENU.

After the LKDOS is initialized, this section of programming is automatically DELETED from the main Daisy program. This "MERGE, USE, and DISCARD" concept is used throughout the Daisy Software to conserve memory for DATA Management.

YOU CAN RE-SET the Initialization ANYTIME during operation, without losing any data. You do this by <BREAK IN with the BREAK KEY and type: <GO TO 2060 ENTER>. There is another Re-Initialization offered at the OFFICE TOOLS menu, but that one CLEARS the data out and loads in the standard Variable Files for a clean program.

IMPORTANT FOR PRINTER  
OPERATION

```

*** Program "Inpt.B6" LLIST *** 3793 Bytes

150 RANDOMIZE USR VAL "24355"
158 LET a=PEEK 24314: LET b=PEEK 24313: LET c=PEE
K 23560: LET a1=PEEK 24311+256*PEEK 24312
160 ON ERR GO TO VAL "252": IF c=12 THEN GO TO
370
162 IF c=13 THEN GO TO 150
164 GO SUB 365
166 GO TO (c=96)*386+(c=7)*310+(c=8)*290+(c=9)*27
5+(c=10)*265+(c=11)*255+(c=12)*370+(c=13)*150+(c=1
4)*392+(c<7)*392
235 POKE 23611,220
238 IF PEEK 23611<221 THEN GO TO 238
240 LET c=PEEK 23560: POKE 23611,220: IF c>31 AND
c<96 THEN GO TO 300
250 GO TO 166
252 ON ERR RESET: GO TO 150
255 LET a=a-1: LET a1=a1-32: IF a<0 THEN LET a=0
: IF a1<b THEN LET a1=b
256 GO TO 301
265 LET a=a+1: LET a1=a1+32: IF a1+64>LEN u$ THEN
BEEP VAL ".2",VAL "14": RANDOMIZE USR VAL "24355
"
266 IF a>13 THEN LET a=14:
267 IF LEN u$(a1-b+1-a*32+481) THEN PRINT AT 0,0;
u$(a1-b+1-a*32 TO);
268 IF a<14 AND LEN u$(a1-b+1-a*32+480) THEN PRIN
T AT 0,0;u$(a1+1-b-a*32 TO a1-b+1-a*32+480);
269 IF a>13 AND LEN u$(a1-b+1-a*32+480+64) THEN P
RINT AT 0,0;u$(a1+1-b-a*32 TO a1+1-b-a*32+480+64);

270 PRINT AT a,b; OVER 1;" "; GO TO 235
275 LET a1=a1+1: LET b=b+1: IF b=32 THEN LET b=0
: LET a=a+1: IF a>21 THEN LET a=a-1: LET b=31: LE
T a1=a1-1
280 IF A1>LEN U$ THEN LET A1=A1-1: LET B=B-0a: I
F B=-1 THEN LET B=31: LET A=A-1
285 PRINT AT a,b; OVER 1;" "; GO TO 235
290 LET b=b-1: LET a1=a1-1: IF b=-1 THEN LET b=3
1: LET a=a-1: IF a=-1 THEN LET b=0: LET a=0: LET
a1=a1+1
295 PRINT AT a,b; OVER 1;" "; GO TO 235
300 PRINT AT a,b;CHR$ c;: LET u$=u$(TO a1)+CHR$
c+u$(a1+1 TO): LET b=b+1: LET a1=a1+1: IF b=32 TH
EN LET b=0: LET a=a+1
301 IF LEN u$(a1-b+1-a*32+481) THEN PRINT AT 0,0;
u$(a1-b+1-a*32 TO);
302 IF LEN u$(a1-b+1-a*32+480) THEN PRINT AT 0,0;
u$(a1-b+1-a*32 TO a1-b+1-a*32+480)
305 PRINT AT a,b; OVER 1;" "; GO TO 235
310 LET d1=(1 AND d1<1)+(2 AND d1>0): LET d2=(a1
AND d1<2)+(d2 AND d1>1): IF d1=2 AND d2<a1 THEN L
ET I$=u$(d2+1 TO a1+1): LET u$=u$(TO d2)+u$(a1+2
TO): LET d2=0: LET d1=0: LET a4=1: GO TO VAL "390
"
312 IF d1=2 AND d2<a1 THEN LET d1=0: LET a4=1: L
ET d2=0: GO TO VAL "390"
335 IF d1=1 THEN PRINT #0;AT 0,0; INVERSE 1;"Pos
ition Cursor to BLOCK END, Then EDIT": PRINT AT
a,b; OVER 1; FLASH 1;"_";: BEEP VAL ".2",15: GO TO
VAL "235"
340 IF d1=2 THEN LET a4=1: GO TO VAL "390"
365 PRINT AT 18,0; INK 5;"ENTER to Resume - Cap+
SYM=HELP *Bik Insert - EDIT=Bik Delete. "; INK 7;
AT a,b; INVERSE 1; OVER 1;"_";: RETURN
370 PRINT AT a,b;" ";
371 IF a1>0 THEN LET u$=u$(TO a1-1)+u$(a1+1 TO
): IF a1>LEN u$ THEN GO TO 150
372 LET b=b-1: LET a1=a1-1: IF b<0 THEN LET a=a-
1: LET b=31: IF a<0 THEN LET a=0
373 IF a1<1 THEN LET a1=1: LET b=0
374 GO TO 301
386 PRINT INVERSE 1; OVER 1;AT a,b;"_"; OVER 0;#0
AT 0,0;"BLOCK INSERT"; INVERSE 0;"<1> To Insert

```

This is The Whole Input And Editing for DATA BASE MANAGEMENT.

Accessed From The :

1. Function menu (Input EDIT)
2. QUICKIE MENU - Line 2926
3. The Sequential Files MANAGER (Line 9668-9737)

During "Turbo I" Mode This group "STAYS ATTACHED" TO DAISY, TO AVOID WAITS FOR MERGE.

Two Other Programs are Attached in TURBO I Mode :

1. Edt.B6 page 16
2. Seqnt.B6 page 8

The Code Program "CAT.C1" is used for FAST Input Typing and Resides in address 24355 + 265

ESCAPE Branches Begin at Line #392

Dont Be Intimidated!

Instead, let Daisy kinda "flow into you" by daily use and often review of the manual. It isn't difficult to get started if you read enough of the manual to understand the concepts. You will be able to do simple operations that you are accustomed to doing with other software. Then everytime you use Daisy you will learn something. And, every time you review the manual you will learn something else. This process will go on almost forever. I, the programmer, learn things from using Daisy that were not planned as functions!! Un-planned benefits occur when so very many planned function overlap in capability.

```

L$ <2> Insert Typing": PAUSE 0: L
ET a3=z: PAUSE 0: LET z=CODE INKEY$-48: LET a4=z:
LET z=a3: IF a4<1 OR a4>2 THEN GO TO VAL "386"
389 IF a4=1 THEN LET u$=u$(TO a1)+I$+u$(a1+1 TO
)
390 IF a4=1 THEN PRINT #0;AT 0,0;"

```

```

": GO SUB 365: GO TO 301
391 PRINT AT 18,0;"Type the Insert, OR ERASE Quot
es and enter String name to Insert.": INPUT LINE m
$: LET u$=u$(TO a1)+m$+u$(a1+1 TO): PRINT AT 18,
0;"

```

```

": PRINT #0;AT 0,0;"
": GO TO VAL "301"

```

```

392 ON ERR GO TO 393: STOP
393 ON ERR RESET : CLS : IF sx=oo THEN PRINT #5
;u$: LET u$="": PRINT AT 0,j,oo;"<1> More TEXT -or-
<2> QUIT": PAUSE NOT PI: LET qs=CODE INKEY$-VAL "
48": IF qs<oa OR qs>ob THEN GO TO VAL "393"
394 IF qs=oa THEN GO TO 150: REM GO SUB VAL "36
5": GO TO VAL "301": IF sx=1 THEN GO TO 9698
395 IF qs=ob THEN GO TO VAL "9687"
396 IF sx=oa THEN GO TO VAL "9698"
398 IF seq=oa THEN GO TO a1t
400 GO TO (xy<oa)*VAL "2950"+(xy=oa)*VAL "3028"+(
xy>oa)*ed

```

\*\*\* Program "edt.B6" LLIST \*\*\* 3061 Bytes

```

2926 ON ERR RESET : RANDOMIZE USR m1: LOAD "ed.C1
"SCREEN$: LET d1=oo: LET sx=oc: LET w$="": LET s$
="": LET xy=ob: LET ep=(ep AND ep>oo AND LEN h$(oa
)>oc)+(og AND ep<oa AND LEN h$(oa)>oc)+(ep AND ep<
oa AND LEN h$(oa)<od): PRINT AT op,ol;LEN u$;" ";
AT op,t3-od;(LEN h$(oa) AND ep<oh)+(LEN I$(oa) AND
ep>og);AT oq,op; FREE ;" ";AT os,od;ep;AT ot,od;
eq
2928 GO SUB 11: GO TO (z>oj)*VAL "2928"+(z=oa)*VAL
"140"+(z=ob)*VAL "2930"+(z=oc)*VAL "3400"+(z=od)*
fm+(z=oe)*VAL "2966"+(z=of)*VAL "3010"+(z=og)*VAL
"3060"+(z=oh)*VAL "2968"+(z=oi)*VAL "2929"+(z=oo)*
VAL "2600"+(z=oj)*VAL "3040"
2929 POKE VAL "23658", (oo AND PEEK VAL "23658">oo)
+(oh AND PEEK VAL "23658"<oa): BEEP oa/od,on: GO T
O VAL "2928"
2930 RANDOMIZE USR m1: LOAD "prt.C1"SCREEN$: LET
xy=oo: PRINT AT oc,ou; INK oe;"EDIT "; INK og: GO
SUB sq: PAUSE oo: LET z=CODE INKEY$-VAL "48": IF z
<oa OR z>ou THEN GO TO ed
2932 IF z<>op THEN LET s$=u$
2933 LET z1=z
2934 IF z1<oh THEN LET u$=h$(z1)
2935 IF z1>og AND z1<op THEN LET u$=I$(z1-og)
2940 IF z1=oa THEN LET u$=a$
2941 IF z1=or THEN LET u$=b$
2942 IF z1=os THEN LET u$=c$
2943 IF z1=ot THEN LET u$=d$
2944 IF z1=ou THEN LET u$=e$
2945 CLS : LET a=oo: LET b=oo: LET al=oo: GO SUB V
AL "365": GO TO m3+oa
2950 IF z1<oh THEN LET h$(z1)=u$
2951 IF z>og AND z1<op THEN LET I$(z1-og)=u$
2954 IF z1=oa THEN LET a$=u$
2956 IF z1=or THEN LET b$=u$
2958 IF z1=os THEN LET c$=u$
2960 IF z1=ot THEN LET d$=u$
2962 IF z1=ou THEN LET e$=u$
2963 IF z1<>op THEN LET u$=s$
2964 LET s$=" "
2965 GO TO ed
2966 LET u$="": GO TO m1+t5
2968 CLS : PRINT AT ou,oo;"Input L$ Data to Insert
Anywhere""during EDIT or Input Typing.": INPUT I
$: INVERSE oo: GO TO ed

```

"Inpt.B6" section is use  
for both MAIN Text DATA  
Bases. Sequential files  
Manager and SMART Text  
db Manager.

The END BRANCH routines for the INPUT - EDIT  
group begins at line #392 which resets ON ERR,  
then Branches to return to the Data Base Manager  
which has been using the Input Edit functions.

This is the EDIT section for the  
Two DATA Bases H\$ and I\$  
ARRAYS. STAYS ATTACHED IF  
TURBO=1

#### Quickie Menu

1. Typing 2. Re-ed 3. Store
4. HELP 5. DELETE BUFFER
6. View/Edit Array Cells
7. Word Replace, Global
8. Input to L\$ for PASTE-UP
9. Cycle Caps 0 Edit Seq DB
- <: Zap a Data Base

Buffer= Cell Limit=  
FREE=

H-0  
I-0

ed

#### Slct Data to EDIT

|            |            |
|------------|------------|
| <1> H\$(1) | <8> I\$(1) |
| <2> H\$(2) | <9> I\$(2) |
| <3> H\$(3) | 10> I\$(3) |
| <4> H\$(4) | 11> I\$(4) |
| <5> H\$(5) | 12> I\$(5) |
| <6> H\$(6) | 13> I\$(6) |
| <7> H\$(7) | 14> I\$(7) |

|            |         |
|------------|---------|
| 15> Typing | 18> C\$ |
| 16> A\$    | 19> D\$ |
| 17> B\$    | 20> E\$ |

21> Abort  
Enter Choice:

```

3010 RANDOMIZE USR m1: LOAD "ve.C1"SCREEN$: LET s
x=oc: LET qs=oc: LET seq=oo: LET xy=oa
3011 GO SUB sq: GO SUB il: LET z1=z: IF z1>oc THEN
GO TO (z=od)*ed+(z>od)*fm
3012 IF z1<oc THEN GO TO VAL "3016"
3013 LET z1=ob: BEEP Pl/oh,on: INPUT "Enter Data D
isk #";dd: RANDOMIZE USR m1: GO TO dd: CLS : RANDO
MIZE USR m1: CAT ".A$": BEEP Pl/oh,os: INPUT "Ent
er NBR only of "; FLASH oa;"t"; FLASH oo;" File to
LOAD"; LINE m$: LET m$="t"+m$+".A$": RANDOMIZE US
R m1: LOAD m$ DATA I$(): BEEP Pl/oh,on: INPUT "Ent
er PGM Disk #";dd: RANDOMIZE USR m1: GO TO dd
3014 CLS : PRINT AT oj,oo;"The Data is now a ""|""
DB FILE and is ready for EDIT""After EDIT you
may RE-SAVE via Menu with an appropriate File #""
""ENTER WHEN READY For EDIT": PAUSE oo: CLS
3016 LET a=(LEN h$(oa) AND z1<ob)+(LEN I$(1) AND z
1>oa): IF a<ob THEN CLS : PRINT #oo;"No Data in t
he Data Base": BEEP ob,on: GO TO ed
3017 BEEP Pl/oh,on: PRINT #oo;"Input Start Cell #
of "; FLASH oa;"H" AND z<ob)+(I" AND z>oa); FLAS
H oo;" DB for View EDIT (1 to 7) ": INVERSE oo: P
AUSE oo: LET y=CODE INKEY$-VAL "48": CLS : IF y<oa
OR y>og THEN GO TO VAL "3015"
3018 FOR n=y TO og: CLS : IF z1=oa THEN PRINT AT
oo,oo;h$(n);
3019 IF z1=ob THEN PRINT AT oo,oo;I$(n)
3020 PRINT #oo;AT oo,oo; FLASH oa;"I" AND z1>oa)+
("H" AND z1<ob);("n;"); FLASH oo; INVERSE oa;"E
NT for NEXT- or-"; INVERSE oo;" "; FLASH oa;"e to
EDIT"; FLASH oo: PAUSE oo: LET t$=INKEY$: IF t$<>"
e" AND t$<>"E" THEN GO TO VAL "3026"
3021 CLS : LET s$=u$: IF z1=oa THEN LET u$=h$(n)
3022 CLS : IF z1=ob THEN LET u$=I$(n)
3024 LET a=oo: LET b=oo: LET al=oo: LET dl=oo: GO
SUB VAL "365": GO TO VAL "301"
3026 NEXT n: GO TO ed
3028 IF z1=oa THEN LET h$(n)=u$
3029 IF z1=ob THEN LET I$(n)=u$
3030 LET u$=s$: LET s$="": LET n=n-oa: NEXT n: GO
TO ed
3040 CLS : PRINT AT oj,ob;"Zap Which DB?""TAB ob;
"[1] H Data Base""TAB 2;""[2] I Data Base""TAB 2;
"[3] Both": GO SUB sq: GO SUB il: IF z=oa THEN DI
M h$(oa): LET ep=oo: LET cx=oo
3042 IF z=ob THEN DIM I$(oa): LET cz=oo: LET eq=o
o
3044 IF z=oc THEN DIM h$(oa): DIM I$(oa): LET cx=
oo: LET cz=oo: LET ep=oo: LET eq=oo
3046 GO TO ed
3400 CLS : IF cx=oo OR cz=oo AND ep>og THEN PRINT
AT om,oa;"Inpt Para Lgth for ";("H$ array" AND cx
=oo)+("I$ array" AND ep=oh AND cx>oo): LET mz=oo
3402 IF cx=oo THEN LET ep=oa: INPUT cx: IF FREE
-cx*og<k4 THEN LET ep=oo: LET cx=oo: GO TO k3+m4
3404 IF cx>oo AND ep>og AND cz=oo THEN LET eq=oa:
INPUT cz: IF FREE -cz*og<k4 THEN LET cz=oo: LET
eq=oo: GO TO k3+m4
3406 IF ep=oa THEN DIM h$(og,cx)
3407 IF ep=oh AND eq=oa THEN DIM I$(og,cz): LET e
q=oa
3408 IF ep<oh THEN LET h$(ep)=u$: LET ep=ep+oa: G
O TO VAL "3411"
3409 IF ep>og AND eq<oh THEN LET I$(eq)=u$: LET e
q=eq+oa
3411 IF eq<oh THEN LET u$=""
3414 GO TO ed
3416 CLS : PRINT AT oj,oo;"Select: ""<1> Daisy DB
Manager""<2> Sequential files Mgr": GO SUB il:
IF z<oa OR z>ob THEN GO TO VAL "3416"
3417 LET alt=(ed AND z=oa)+(VAL "9668" AND z=ob):
IF turbo<>oa THEN LET turbo=oa: GO SUB VAL "495":
RANDOMIZE USR m1: MERGE "tb.B6": GO TO VAL "850"
3418 GO TO alt

```

#### View Edit Options

- [1] View Edit "H" DB (mem)
- [2] View Edit "I" DB (mem)
- [3] LOAD/Edit Seq Files
- [4] To Quickie Menu
- [5] To Function Menu

These various Edit  
 Functions Operate  
 By individual  
 Selections from  
 The Quickie Menu -  
 (line 2926)  
 (goto Ed) gets The  
 Quickie Menu  
 from a BREAK  
 STATUS.



\*\*\* Program "seqnt.B6" LLIST \*\*\*

2780 Bytes

This is the Sequential Files DATA  
BASE MANAGER. STAYS ATTACHED  
IF TURBO = 1

```

9668 RANDOMIZE USR m1: LOAD "seq1.C1"SCREEN$: LET
sx=oo: LET y=oo: LET pt=oo
9669 PAUSE oo: LET z=CODE INKEY$-VAL "48": GO TO (
z<oc)*VAL "9670"+(z>od)*VAL "2604"+(z=oc)*VAL "969
0"+(z=od)*VAL "9690"+(z>od)*VAL "2604"
9670 LET u$="": CLS : PRINT AT ou,oo;"Turn Disk ON
, Enter Data Disk # ": INPUT dd: RANDOMIZE USR m1:
GO TO dd: CLS : RANDOMIZE USR m1: CAT ".A$": INP
UT "ENTER A File # to OPEN";zz: CLS : LET y$="Dta"
+STR$ zz+ ".A$"+ " " OUT ": RANDOMIZE USR m1: OPEN
#5,y$: GO TO (z=oa)*VAL "9671"+(z=ob)*VAL "150"
9671 LET a=ov: LET b=oo: PRINT AT a,b;"_";
9672 POKE 23611,220
9675 IF PEEK 23611<221 THEN GO TO 9675
9676 LET c=PEEK 23560: POKE 23611,220
9677 IF c=12 THEN GO TO 9683
9678 IF c=13 THEN GO TO 9688
9679 IF c=14 THEN GO TO 9685
9680 PRINT AT 0,29;LEN u$;AT a,b;CHR$ c;: LET b=b+
1: IF b=32 THEN LET b=0: LET a=21: POKE 23692,255
: PRINT " " ";
9681 PRINT AT a,b;"_";: LET u$=u$+CHR$ c
9682 GO TO 9672
9683 PRINT AT a,b;" ": LET u$=u$(TO LEN u$-1): L
ET b=b-1: IF b=-1 THEN LET b=31: LET a=a-1
9684 PRINT AT a,b;" ": GO TO 9672
9685 PRINT #5,u$: LET u$=" "
9686 CLS : PRINT AT ou,oo;"<1> Create More Text" " "
<2> QUIT": INPUT xx: CLS : IF xx=oa THEN LET b=oo
: GO TO VAL "9672"
9687 RANDOMIZE USR m1: CLOSE #5: CLS : INPUT "Ente
r Program Disk #:";dd: RANDOMIZE USR m1: GO TO dd:
GO TO VAL "9668"
9688 LET b=0: POKE 23692,255: PRINT "AT a,b;"_";:
PRINT #5,u$: LET u$="": GO TO 9672
9689 RETURN
9690 RANDOMIZE USR m1: CLOSE #5: RANDOMIZE USR m1:
LOAD "seq2.C1"SCREEN$: LET sx=oa
9691 PAUSE oo: LET z=CODE INKEY$-VAL "48": IF z>od
THEN GO TO VAL "9668"
9692 PRINT AT ou,ob; INVERSE oa;"Install Data Disk
, ENTER DD #": INVERSE oo: INPUT y: RANDOMIZE USR
m1: GO TO y: CLS : RANDOMIZE USR m1: CAT ".A$": I
NPUT "ENTER Whole File Name to OPEN ":x$: LET x$
=x$+" IN ": RANDOMIZE USR m1: OPEN #5,x$: CLS : GO
TO (z=oa)*VAL "9700"+(z=ob)*VAL "9693"+(z>oc)*VAL
"9706"+(z=od)*VAL "9706"+(z>od)*VAL "9736"
9693 IF FREE <VAL "12000" THEN BEEP oa,ob: RANDO
MIZE USR m1: CLOSE #5: GO TO VAL "9668"
9694 DIM k$(ol,m8): FOR n=oa TO ol: ON ERR GO TO
9695: INPUT #5; LINE k$(n): NEXT n: ON ERR GO TO
VAL "9696": STOP
9695 ON ERR RESET : ON ERR GO TO VAL "9696": STO
P
9696 ON ERR RESET : LET mx=n-oa: RANDOMIZE USR m1
: CLOSE #5: LET x$=x$(TO LEN x$-ob): RANDOMIZE US
R m1: ERASE x$, : LET x$=x$+" OUT ": RANDOMIZE USR
m1: OPEN #5,x$
9697 FOR n=oa TO mx: LET u$=k$(n): GO SUB VAL "365
": GO TO VAL "301"
9698 PRINT #5,u$: LET u$="": NEXT n: RANDOMIZE USR
m1: CLOSE #5: DIM k$(oa): INPUT "ENTER PROGRAM DI
SK #:";dd: RANDOMIZE USR m1: GO TO dd:: GO TO VAL
"9668"
9699 LET u$="": RANDOMIZE USR m1: CLOSE #5: INPUT
"ENTER ProgramDisk #:";dd: RANDOMIZE USR 100: GO T
O dd: GO TO VAL "9668"
9700 CLS :: FOR n=oa TO m5: ON ERR GO TO 9702: IN
PUT #5; LINE u$: PRINT n;TAB oe;" ENTER For Next--
-----"u$: LET u$="": PAUSE oo: NEXT n
9701 PRINT u$' INVERSE oa;n;TAB oe;" ENTER For Nex
t": PAUSE oo: NEXT n
9702 ON ERR RESET : ON ERR GO TO VAL "9703": STO
P
9703 ON ERR RESET : RANDOMIZE USR m1: CLOSE #5: I
PUT "ENTER Program Disk #:";dd;: RANDOMIZE USR 10

```

#### Sequential File Manager

- [1] Create Seq File of Lines
- [2] Create Seq File of Paragraphs
- [3] View - Edit File
- [4] PRINT FILE
- [5] Return to Daisy Menu

The Sequential File  
DATA BASE Manager is  
supported By the  
"INPT.B6" Section  
and the  
"edT.B6" Section.

The word processor "WdPro.B6"  
is merged when <3> (Below)  
is elected.  
Once print out is  
finished "WdPro.B6"  
is deleted.

#### Seq File View- Edit- Print

- [1] View a File
- [2] Edit a Para File
- [3] PRINT a Para File
- [4] PRINT a Line File
- [5] To Main Menu

```

0: GO TO dd: GO TO VAL "9668"
9708 RANDOMIZE USR m1: LOAD "seq.C1"SCREEN$
9709 PAUSE oo: LET y=CODE INKEY$-VAL "48": IF y>ob
THEN GO TO (y=oc)*VAL "2604"+(y=oc)*VAL "9668"
9710 PAUSE oo: IF y=ob THEN GO TO VAL "9722"
9712 CLS : INPUT "INPUT Left Margin:";mar: RANDOMI
ZE USR m1: POKE VAL "16094",mar: LPRINT : INPUT "L
ines per Page?";ln: INPUT "Start print at Line #?"
;st: INPUT "Page #?";pg
9714 RANDOMIZE USR m1: OPEN #5,x$: FOR n=oa TO m6:
LET u$="": ON ERR GO TO VAL "9718": INPUT #5; LI
NE u$: LPRINT u$: GO SUB sp
9716 NEXT n
9718 ON ERR RESET : ON ERR GO TO VAL "9720": STO
P
9720 ON ERR RESET : RANDOMIZE USR m1: CLOSE #5: G
O TO VAL "9668"
9722 RANDOMIZE USR m1: POKE VAL "16090",lx: RANDOM
IZE USR m1: POKE VAL "16094",oo: LPRINT : IF turbo
=oo THEN RANDOMIZE USR m1: MERGE "wdpro.B6"
9724 RANDOMIZE USR m1: OPEN #5,x$: FOR n=oa TO ol:
ON ERR GO TO VAL "9730": INPUT #5; LINE w$: GO S
UB te: GO SUB sp
9726 NEXT n
9730 ON ERR RESET : ON ERR GO TO VAL "9732": STO
P
9732 ON ERR RESET : RANDOMIZE USR m1: CLOSE #5: I
F turbo=oo THEN DELETE VAL "3742",VAL "3812"
9734 GO TO VAL "9668"
9735 GO SUB VAL "365": GO TO VAL "301"
9736 RANDOMIZE USR 100: CLOSE #5: INPUT "Enter PRO
GRAM DISK #:";dd: RANDOMIZE USR m1: GO TO dd: GO T
O VAL "9668"
9866 DELETE VAL "9822",VAL "9864": GO TO fm

```

#### INFORMATION . . .

Printing of LINE Files is  
to your INPUT TAB Position.

Printing of PARAGRAPH Files  
is to the PRE-SET FORMAT of  
the Word Processor.

You may Abort to set the  
Format Desired. . .

<1> Print Line File  
<2> Print Paragraph File  
<3> To Daisy Function Menu  
<4> To Seq Menu

ENTER CHOICE

Word Processor

\*\*\* Program "repp.B6" LLIST \*\*\* 1466 Bytes

```

2100 RANDOMIZE USR m1: LOAD "rp.C1"SCREEN$: LET p
c=oo: LET az=oo: LET no=oo: LET zl=oo: LET rpr=oa:
LET pt=oo: LET ms=oo: LET mx=oo: LET ln=oo: LET g
g=oa: LET mm=oo
2101 GO SUB sq: GO SUB ll: GO TO (z<oa OR z>oh)*VA
L "2101"+(z>oo AND z<od)*VAL "2102"+(z=od)*fm+(z=o
f)*k2+(z=oe)*VAL "2174"+(z=og)*VAL "8990"+(z=oh)*V
AL "2149"
2102 IF z<oc AND turbo=oo THEN RANDOMIZE USR m1:
MERGE "usrpgm.B6": GO TO VAL "2103"
2103 IF z=ob THEN RANDOMIZE USR m1: MERGE "lh.B6"
: GO TO VAL "2104"
2104 IF z>oa THEN INPUT "Key: <1> for Manual Addr
essing <2> for Mail List Adr";mm: CLS : IF z
=ob THEN GO SUB VAL "2590": GO SUB VAL "4055"
2105 IF z=oc THEN CLS : PRINT AT os,ob;"Do you wa
nt Auto Page Control? <1> yes or <2> No ": INPUT
pc: CLS : PRINT AT os,ob;"Twin Labels? <1> Yes - <
2> No": INPUT tw: CLS
2106 IF z=oa OR mm=oa THEN INPUT "ENT nbr copies
";no: LET ms=oa: LET mx=no: IF mm=oa AND z=oc THEN
GO SUB VAL "2170"
2108 IF z=oc THEN INPUT "ENT TAB ";tz: INPUT "ENT
sps";sx: CLS
2110 IF mm=ob THEN INPUT "ENT ml strt nbr ";ms: I
NPUT "ENT end ml nbr ";mx: CLS
2112 FOR l=ms TO mx
2116 IF z=ob THEN GO SUB VAL "4061": CLS
2118 IF z=oc THEN GO SUB VAL "2150"
2120 IF z<oc THEN GO TO VAL "2180"
2138 NEXT l
2139 LET lh=oo
2140 IF zm<mz THEN LET zm=zm+oa: GO SUB k9+oj: GO
TO VAL "2112"
2144 LET z2=oo
2148 GO TO VAL "2410"
2149 GO SUB VAL "8150": LET gg=oo: GO TO VAL "2410"
2150

```

Page

The MAIN PRINTING MANAGEMENT  
Group. STAYS ATTACHED IF TURBO=2  
MERGES IF TURBO=1 OR 0.  
Two other groups STAY ATTACHED IF  
TURBO=2 -  
<1> USR Pgm.B6 <2> WdPro.B6

#### Printing Menu

<1> Manuscript  
<2> Letters/Invoices  
<3> Labels/Envelopes  
  
<4> To Function Menu  
<5> Postscript ON  
<6> To Mailing List  
  
<7> Print Disk Files  
<8> Print Outline  
<9> Print Disk Catalog  
  
<0> Print Screens  
<:> Utilities Menu

2100

```

2152 IF mm=oa THEN LPRINT TAB tz;p$(oa)'TAB tz;p$(ob)'TAB tz;p$(oc)'TAB tz;p$(od): LET nn=nn+od: LET a=oo: GO SUB VAL "2164"
2154 IF mm=ob THEN FOR n=oa TO od: LPRINT TAB tz;o$(1,n)': NEXT n: LET nn=nn+od: LET a=oo: GO SUB VAL "2164"
2160 IF tw=oa THEN LPRINT TAB tz;n$(of)'TAB tz;n$(og)'TAB tz;n$(oh)': LET nn=nn+od: LET a=oo: GO SUB VAL "2164"
2161 IF pc=oa AND nn>=ln-od THEN GO SUB pe
2162 RETURN
2164 IF a<sx THEN LPRINT ': LET a=a+oa: IF pc=oa THEN LET nn=nn+oa
2166 IF a<sx THEN GO TO VAL "2164"
2167 IF pc=oa AND nn>=ln-od THEN GO SUB pe
2168 LET a=oo: IF pc<>oa THEN LET nn=oo
2169 RETURN
2170 DIM p$(od,VAL "26"): INPUT "First Line: ";p$(oa): INPUT "Second line: ";p$(ob): INPUT "Third Line: ";p$(oc): INPUT "Fourth Line: ";p$(od): RETURN
2172 DIM p$(oa): RETURN
2174 BEEP oa/ob,ov: LET scp=oa: GO TO VAL "2101"

```

THE Built-In MAIL LIST PRINTER: Line 2150 begins this utility. The Mailing List Data Base is printed via selection <3> at the Print Management Menu. Mail List records are 7 lines. All 7 lines of each of the records may be printed in a File Listing, or the first four lines may be printed for labels or envelopes. Such print products may be "of a memory file", or "of disk files" when "<9> Print Disk Files" has been selected at the Print Management Menu. In the latter case, the file numbers of selected disk files are automatically loaded and printed as either lists, to labels, or to envelopes.

\*\*\* Program "usrpgm.B6" LLIST \*\*\*

*998 Bytes / USER programming group. STAYS ATTACHED IF Turbo = 2.*

```

2182 LET w$=h$(1): GO SUB tel: GO SUB sp
2184 LET w$=h$(2): GO SUB tel: GO SUB sp
2187 LET w$=h$(3): GO SUB tel: GO SUB sp
2188 LET w$=h$(4): GO SUB tel: GO SUB sp
2190 LET w$=h$(5): GO SUB tel: GO SUB sp
2192 LET w$=h$(6): GO SUB tel: GO SUB sp
2194 LET w$=h$(7): GO SUB tel: GO SUB sp
2196 IF zy<oa THEN GO TO 2206
2197 REM * Disk Sequencer * Do not disturb lines
2198 - 2206
2198 IF zy>oo AND zx<zy THEN LET zx=zx+oa: LET az=oo: LET m$=STR$ zx: LET m$=m$+".A$": RANDOMIZE USR m1: GO TO dd: RANDOMIZE USR m1: LOAD m$ DATA h$(): RANDOMIZE USR m1: GO TO pd: GO TO VAL "2182"
2200 IF az=oa THEN LET m$=STR$ zp+".A$": RANDOMIZE USR m1: GO TO dd: RANDOMIZE USR m1: LOAD m$ DATA h$(): RANDOMIZE USR m1: GO TO pd: LET az=oo: LET zx=zp
2204 LET az=oo
2206 ON ERR GO TO VAL "2280": IF I$(oa,oe)>"" THEN ON ERR GO TO VAL "2208": STOP
2208 ON ERR RESET
2210 LET w$=i$(1): GO SUB tel: GO SUB sp
2212 LET w$=i$(2): GO SUB tel: GO SUB sp
2214 LET w$=i$(3): GO SUB tel: GO SUB sp
2216 LET w$=i$(4): GO SUB tel: GO SUB sp
2218 LET w$=i$(5): GO SUB tel: GO SUB sp
2220 LET w$=i$(6): GO SUB tel: GO SUB sp
2222 LET w$=i$(7): GO SUB tel: GO SUB sp
2280 ON ERR RESET
2300 IF lh>oo THEN GO SUB VAL "4090": GO SUB pe: GO TO VAL "2138"
2399 REM ** END of User Program Group. Do not disturb AFTER.
2400 IF spl=oa THEN GO SUB pe: LET nn=oo: GO TO VAL "2138"
2401 IF lh>oo THEN LET z=ob
2402 IF z=ob THEN GO SUB k4++9: LET z2=z2+oa: IF lh>oo THEN LET z=oa
2403 GO SUB pe: LET nn=oo: GO TO VAL "2138"

```

*Schedules The Printing of The "H AND I" data bases.*

*Edit These Lines to Schedule Other DATA PRINTING. Sequence As desired*

*The Disk Sequencer. Loads in Disk Files for Printing IF "PRINT Disk Files" selected. Loops BACK TO Line 2182 To Keep on Printing Disk Files. - Ignored IF "Disk Files printing" NOT Selected.*

*Change PRINT Sequence As desired OR Add other PRINTING.*

*End Routines - This whole group OPERATES under control of The "L counter" Line #2112 of The "repp.B6" program.*

\*\*\* Program "wdpro.B6" LLIST \*\*\* 1371 Bytes

The Daisy Word Processor

```

3742 LET nt=nz: IF nt=1 THEN LET ll=ll-5: LET tb=
tb+5
3743 IF LEN w$<4 THEN GO TO 3788
3744 IF w$(1 TO 3)=" " THEN GO TO 3810
3745 IF LEN w$<=11 AND w$(LEN w$-2 TO LEN w$)="
" THEN LET w$=w$(TO LEN w$-2): GO TO 3748
3746 IF LEN w$<=11 THEN GO TO 3748
3747 IF LEN w$>=11 AND w$(11-2 TO 11)=" " THEN
LET w$=w$(TO 11-2)
3748 IF w$(1)=" " THEN LET w$=w$(2 TO): GO TO 37
46
3749 IF LEN w$<=11 THEN GO TO 3800
3751 LET m$=w$(TO 11)
3753 IF m$(LEN m$)<>" " AND w$(LEN m$+1)=" " THEN
LET w$=w$(LEN m$+1 TO): GO TO 3758
3754 IF m$(LEN m$)<>" " AND w$(LEN m$+1)<>" " THEN
LET m$=m$(TO LEN m$-1): GO TO 3754
3756 LET w$=w$(LEN m$+1 TO)
3758 IF m$(LEN m$)=" " THEN LET m$=m$(TO LEN m$-
1): GO TO 3758
3759 IF jy=oa THEN GO SUB 3774
3760 LPRINT TAB tb;m$: IF nt=1 THEN LET nt=0: LET
ll=ll+5: LET tb=tb-5
3763 LET nn=nn+1: IF sd=2 THEN GO SUB sp
3764 IF nn>=ln THEN GO SUB pe
3768 GO TO 3743
3774 LET pp=LEN m$: IF pp=11 THEN RETURN
3776 FOR r=1 TO pp: IF m$(r)=" " THEN GO TO 3780
3778 NEXT r: GO TO 3774
3780 LET m$=m$(TO r)+" "+m$(r+1 TO pp): LET pp=LE
N m$: IF pp<>11 THEN LET r=r+1
3784 IF pp<>11 THEN GO TO 3778
3786 RETURN
3788 IF LEN w$<3 THEN GO TO 3810
3790 IF w$(1 TO 3)=" " THEN GO TO 3810
3792 IF w$(1)=" " THEN LET w$=w$(2 TO): GO TO 37
92
3794 IF LEN w$<=11 THEN GO TO 3800
3795 IF w$(LEN w$)=" " THEN LET w$=w$(TO LEN w$-
1): GO TO 3794
3796 GO TO 3751
3800 LPRINT TAB tb;w$: IF nt=1 THEN LET ll=ll+5:
LET tb=tb-5: LET nt=0
3802 LET nn=nn+1: IF sd=2 THEN GO SUB sp
3804 IF nn>=ln THEN GO SUB pe
3810 LET w$="": IF nt=1 THEN LET ll=ll+5: LET tb=
tb-5
3812 IF gg=oa THEN RETURN

```

\*\*\* Program "pe.B6" LLIST \*\*\* 541 Bytes

```

5900 IF pr=oa THEN LPRINT "''";TAB qq;pg;"''''''
''': LET nn=oa: LET pg=pg+oa: RETURN
5901 IF nn<ln+ob THEN LPRINT : LET nn=nn+oa: IF n
n<ln+ob THEN GO TO VAL "5901"
5902 IF cp>oa THEN GO TO VAL "5906"
5903 LPRINT TAB qq;pg: LET nn=oa: LET pg=pg+oa: OU
T po,ol: IF pr=oa THEN GO TO VAL "5913"
5904 GO SUB VAL "5912": GO TO VAL "5913"
5906 IF cp=ob THEN LPRINT TAB lx/ob;pg: LET pg=pg
+oa
5908 LET nn=oa: GO SUB VAL "5912"
5909 LET cp=(oa AND cp=ob)+(ob AND cp=oa): IF cp=o
a THEN LET qq=(VAL "34" AND pr=ob)+(VAL "36" AND
pr=oa): LET tb=oj
5910 IF cp=ob THEN LET qq=(VAL "92" AND pr=ob)+(V
AL "99" AND pr=oa): LET tb=(VAL "66" AND pr=ob)+(V
AL "71" AND pr=oa)
5911 GO TO VAL "5913"
5912 BEEP oa,ov: CLS : PRINT AT oj,ob;"Feed New Sh
eet, then ENTER": PAUSE oa: LET nn=oa: CLS : RETUR
N

```

The Daisy Word processor is a simple "line formatter-printer". (And most users think of Smart Text as a WORD PROCESSOR.). The Word processor will print anything that it sees "In W\$". You can even use it in the Direct Mode. Try this: <LET W\$="Keep it simple stupid!"> <GO TO TE ENTER>.

So, the key is what you put into "W\$" and send to the line #3742 via <GO TO or GO SUB te>.

The formatting is done via "variable switches" that turn ON or OFF certain functions, and assign Line Length (ll) and TAB (tb). These variable are managed by the Smart Text Menu selections. At line 3742, if nz=1 then you have elected "first line indent". Then the "1" is transferred to var "nt" and the line length is reduced by five characters (LL). Then the W\$ data is processed. Traps are to prevent EMPTY string from reaching the following formatter lines. Then traps prevents SHORTER than full lines of data from being processed. Any line shorter than line length "LL" is printed at line #3800.

Line 3748 trims off any spaces at the beginning of the W\$ data group. Line 3747 detects "Sentence structure" and then line 3749 sends a sentence line to line 3800 to be printed. Line 3751 begins to "nibble" W\$ into "line length streams of characters". Each nibble reduces W\$ by a line length and that data is placed into m\$. The processing of M\$ results in trimming of leading spaces, then trailing spaces. Then any "partial word" is trimmed off and put back into the main text (w\$). The result is a group of characters in m\$ that may not be a whole line of characters. If not, then 3759 evaluates "jy". If jy=1 then you have elected "right margin justify" and the m\$ characters are sent to be justified at the routine beginning at 3774.

Spaces are inserted between words to push the last character out to make LEN M\$=11. Then line 3760 prints the whole line of characters, beginning at TAB tb. If it is the "first line, and INDENT is in force, then indent is cancelled (LET nt=0:LET ll=ll+5:LET tb=tb+5). Line 3762 increments the line counter nn, and evaluates "sd". If sd=1 then Double Spacing is in effect and a line space is printed (GO SUB sp). Line 3764 evaluates for page length. If nn (line counter) is at the pre-set page length (ln) then a page end is directed by <GO SUB pe>. Line 3768 then returns to the beginning to nibble off another "ll length" group of characters from the text carrier w\$.

The process continues until the last line of characters is LESS than ll in length, upon which time line 3746 sends the end line to line 3768 for printing. Then all data in w\$ has been printed, and w\$ has been eliminated. The clean-up routine follows at line 3802. Line 3812 evaluates "gg". If gg=1 then "sequenced printing" is in progress. A RETURN is to the User Program Group to pick up the next W\$ data group, upon which the W\$ group is again sent to line TE (3742) for printing. If sequenced printing is NOT in effect, then gg=0, and line #3814 executes to GO TO the function menu.

\*\*\* Program "lh.B6" LLIST \*\*\*

780 Bytes

```

4055 INPUT "ENT Date: ";n$(oj): IF mm=ob THEN RET
URN
4056 INPUT "1st Name ";n$(oe)
4058 CLS : PRINT AT oj,oo;"(Optional: Omit line wi
th ENTER)": INPUT "Co Name ";n$(oa): INPUT "Dept "
;n$(ob): INPUT "St adr ";n$(oc): INPUT "Cty,St,Zip
";n$(od): CLS
4059 IF gg=oa THEN RETURN
4060 LET nn=oo
4061 IF lh=oa AND pg<ob THEN FOR j=oa TO ok: GO
SUB sp: NEXT j: LPRINT TAB qq-oh;n$(oj): LET nn=nn
+oa: GO SUB sp: GO TO VAL "4073"
4062 FOR t=of TO oj
4064 IF n$(t)(oa)=" " THEN NEXT t
4066 LET m$=n$(t)
4067 IF m$(LEN m$)=" " THEN LET m$=m$(TO LEN m$-
oa): GO TO VAL "4067"
4068 LET pg=oa: IF m$(oa)<>" " THEN LPRINT TAB qq
-LEN m$/ob;m$: LET nn=nn+oa
4069 IF t=oj AND spl=oa THEN RETURN
4070 IF t=oj THEN GO SUB sp
4071 IF t=oj THEN GO TO VAL "4073"
4072 NEXT t
4073 LET a=oa
4074 IF mm<ob THEN LET m$=n$(a)
4076 IF mm=ob THEN LET m$=o$(1,a)
4078 IF a=oe THEN LET m$="Dear "+m$
4080 IF a=oe THEN GO SUB sp: LPRINT TAB tb;m$: LE
T nn=nn+oa: GO SUB sp: IF gg=oa THEN RETURN
4082 IF a=oe THEN GO TO VAL "3956"
4084 IF m$(oa)=" " THEN LET a=a+oa: GO TO VAL "40
74"
4086 IF m$(oa)<>" " THEN LPRINT TAB tb;m$: LET nn
=nn+oa
4087 LET a=a+oa
4088 GO TO VAL "4074"

```

Letter Support group  
Merges AS Needed.  
Handles either MANUAL input  
OR MAILING LIST Merge of  
NAMES, ADDRESSES.  
FORMATS And PRINTS Letterhead,  
Business address Block,  
Salutation.

### FORMAT MENU

Select until Finished

- <1> Single Spc
- <2> Dbl Spc
- <3> Print Style Menu
- <4> Block Indent Off
- <5> Block Indent ON
- <6> To Fun Menu
- <7> Column Print
- <8> Let Head Strry

2083

\_\_\_Pg Cent=\_\_\_Pg Lgth=

| Max<br>Line | Line<br>Lgth | Match<br>Line | Line=<br>Page=<br>Tab = |
|-------------|--------------|---------------|-------------------------|
|-------------|--------------|---------------|-------------------------|

\*\*\* Program "fomat.B6" LLIST \*\*\*

1772 Bytes

```

2080 RANDOMIZE USR m1: LOAD "fo5.C1"SCREEN$: LET
mat=11: LET xo=1x: LET lo=11: LET rs=1d: LET cp=oo
: LET lh=oo
2081 PRINT AT ol,VAL "24";"id=";ld;" ";AT oe,os;(
"a" AND sd<ob)+(" " AND sd>oa);AT of,os;("a" AND s
d>oa)+(" " AND sd<ob);AT op,ob;("Pica " AND ps<oc)
+("Elite" AND ps=oc)+("Cond " AND ps>oc);AT ot,ob;
1x;AT op,oq;qq;AT op,VAL "28";ln;AT ot,ol;11;AT ot
,oq;mat;AT or,VAL "28";nn;AT os,VAL "28";pg;AT ot,
VAL "28";tb: GO SUB sq: GO SUB 11: IF z<oa OR z>oh
THEN GO TO VAL "2081"
2082 IF z=og THEN LET ps=od: LET 1x=(VAL "126" AN
D pr>oa)+(VAL "136" AND pr<ob): LET 11=(VAL "50" A
ND pr>oa)+(VAL "55" AND pr<ob): CLS : PRINT AT oj,
ob;"Start Col 1 or 2 ?": INPUT "key 1 or 2";cp: IF
cp=oa THEN LET qq=(VAL "34" AND pr>oa)+(VAL "36
AND pr<ob"): LET tb=oj
2083 IF z=og AND cp=ob THEN LET qq=(VAL "90" AND
pr=ob)+(VAL "99" AND pr=oa): LET tb=(VAL "64" AND
pr=ob)+(VAL "70" AND pr=oa)
2084 LET sd=(z AND z<oc)+(sd AND z>ob): IF z=od TH
EN LET tb=(tb-oe AND 11<1x-ol)+(tb AND 11>1x-oj):
LET 11=(11+oj AND 11<1x-ol)+(11 AND 11>1x-oj): LE
T 1d=1d-oj: BEEP oa/od,ol: PRINT AT ot,ol;11;AT ot
,oq;mat;AT ot,VAL "28";tb;" ": GO TO VAL "2081"
2085 IF z=oe THEN LET tb=(tb+oe AND 11>ou)+(tb AN
D 11<VAL "21"): LET 11=(11-oj AND 11>ou)+(11 AND 1
1<VAL "21"): LET 1d=1d+oj: BEEP oa/od,ol: PRINT AT
ot,ol;11;AT ot,oq;mat;AT ot,VAL "28";tb;" ": GO T
O VAL "2081"
2086 GO TO (z<oc)*VAL "2081"+(z=oc)*VAL "2093"+(z>
oc)*VAL "2088"
2088 BEEP oa/od,os: LET lh=(oa AND z=oh)+(oo AND z
<>oh): IF cp<oa AND pr=VAL "1" THEN LET 1x=(VAL "
134" AND ps>VAL "3")+ (VAL "96" AND ps=VAL "3")+ (VA

```

The FORMAT SECTION, lines 2080-2081, MERGES when FORMAT is selected at the Function Menu. This group MERGES in the Style Menu if needed for Printer Style Changes (lines 8001 thru 8110). The FORMAT MENU (above) displays all of the needed information about Print Style, and the existing Printing Format. The numbered electives allow for a plethora of FORMAT changes. The PROMPTS after allow the standard Daisy CENTER PRINT format, or OFFSET printing, or Column Print. BLOCK INDENT <ON-OFF> selection allows text to be printed "Indented 5 characters" or "out-dented five characters" from both margins. COLUMN PRINT selection allow the print to start at either Column 1 or Column 2, and at top of page or any line number down from the top. The "PAGE END" routine does the "column switching". The Column Print Mode is most valuable when using Cut Sheets of paper to print two formatted columns.



```

L "80" AND ps<VAL "3"): REM Dot Matrix
2089 IF cp<oa AND pr=VAL "2" THEN LET ix=(VAL "12
4" AND ps>VAL "3")+(VAL "96" AND ps=VAL "3")+(VAL
"80" AND ps<VAL "3"): LET qq=INT (ix/ob): REM Dais
y
2090 CLS : PRINT AT oj,ob;"INPUT Printer Line Spa
cing""<1> 1/8""=76 Lines;""<2> 11/64""=56 Line
s""<3> 3/16""=50 Lines""<4> 1/4""=37 Lines": I
NPUT x: LET ln=INT (VAL ".6"+(VAL "9.5"/VAL ".125"
AND x<ob)+(VAL "9.5"/VAL ".172" AND x=ob)+(VAL "9
.5"/VAL ".187" AND x=oc)+(VAL "9.5"/VAL ".25" AND
x>oc)): CLS : PRINT AT oj,ob;"Set Printer for Line
Spacing""TAB ob;"Then Enter Lines per page?""TA
B ob;"Current page lgth=";ln: INPUT ln: CLS : INPU
T "Enter Page#";pg: INPUT "Enter Line Count start"
;nn: IF cp>oa THEN GO TO VAL "2092"
2091 CLS : PRINT AT oj,oo;"Line length=";ll;"Matc
h Line=";mat: INPUT "New Line Length?";ll: LET q
q=INT (ix/ob): LET tb=INT ((ix-ll)/ob): PRINT "Tab
=";tb: INPUT "New Tab?";tb

```

\*\*\* Program "stymn.B6" LLIST \*\*\* 1899 Bytes

```

8001 RANDOMIZE USR m1: LOAD "ps.C1"SCREEN$
8002 LET it=oo: LET px=ps: LET xo=ix: LET lo=ll: G
O SUB sq: GO SUB ll: IF z<oa OR z>og THEN GO TO k
8+ob
8004 IF z=og THEN GO TO VAL "8114"
8005 IF pr=ob AND z<oe THEN GO TO VAL "8018"
8006 IF pr=ob AND z>od THEN GO TO VAL "8002"
8008 IF z=of THEN LET ix=INT (ix/ob+oa/ob): LET q
q=ix/ob: GO SUB VAL "8016": LET ma=INT (oa/ob+ix*1
o/xo): GO SUB VAL "8014": LET tb=INT ((ix-ll)/ob+o
a/ob): BEEP oa/ob,ov: GO TO VAL "8114"
8010 IF z=oe THEN LET it=oa: GO SUB VAL "8110": B
EEP oa/ob,ov: GO TO VAL "8114"
8012 LET ps=z: LET ix=(t8 AND ps<oc)+(t9+of AND ps
=oc)+(VAL "136" AND ps=od): LET qq=INT (ix/ob+oa/o
b): LET ma=INT (oa/ob+ix*1o/xo): GO SUB VAL "8014"
: LET tb=INT ((ix-ll)/ob+oa/ob): GO TO VAL "8024"
8014 CLS : PRINT AT oj,ob;"Key In Line Length""TA
B ob;lx;"=Max ";ma;"=Match Margins": INPUT "Inpu
t Line Length ";ll: CLS : RETURN
8016 GO SUB bu: OUT po,zo: GO SUB bu: OUT po,t8+og
: GO SUB bu: OUT po,oa: GO SUB bu: OUT po,VAL "155
": GO SUB bu: OUT po,t8+og: GO SUB bu: OUT po,oa:
GO SUB bu: OUT po,oy: RETURN
8018 LET ps=z: LET ix=(VAL "84" AND ps<oc)+(VAL "9
8" AND ps=oc)+(VAL "126" AND ps=od): LET qq=INT (l
x/ob+oa/ob): LET ma=INT (oa/ob+ix*1o/xo): GO SUB V
AL "8014": LET tb=INT ((ix-ll)/ob+oa/ob): GO TO VA
L "8114"
8024 GO SUB mt: GO TO VAL "8114"
8030 REM ** The Dot Matrix Printer Control **
8031 IF pr<>oa THEN RETURN
8034 GO SUB bu: OUT po,zo: GO SUB bu: OUT po,m1+ou
: GO SUB bu: OUT po,oo: GO SUB bu: OUT po,VAL "155
": GO SUB bu: OUT po,m1+ou: GO SUB bu: OUT po,oo:
GO SUB bu: OUT po,zo: GO SUB bu: OUT po,oe+t4: GO
SUB bu: OUT po,oo: GO SUB bu: OUT po,VAL "155": GO
SUB bu: OUT po,oe+t4: GO SUB bu: OUT po,oo: GO SU
B bu: OUT po,zo: GO SUB bu: OUT po,VAL "87": GO SU
B bu: OUT po,oo: GO SUB bu: OUT po,VAL "155": GO S
UB bu: OUT po,VAL "87": GO SUB bu: OUT po,oo: GO S
UB bu: OUT po,os: GO SUB bu: OUT po,VAL "146": GO
SUB bu: OUT po,zo: GO SUB bu: OUT po,oc+t5: GO SUB
bu: OUT po,VAL "155": GO SUB bu: OUT po,oc+t5: GO
SUB bu: OUT po,oy
8035 GO SUB bu: OUT po,zo: GO SUB bu: OUT po,t6+ol
: GO SUB bu: OUT po,VAL "155": GO SUB bu: OUT po,t
6+ol: GO SUB bu: OUT po,oy: GO SUB bu: OUT po,zo:
GO SUB bu: OUT po,t8: GO SUB bu: OUT po,VAL "155":
GO SUB bu: OUT po,t8: GO SUB bu: OUT po,oy
8036 IF it=oa THEN GO SUB VAL "8110"

```

## Print Style And Printer Control Section. Menu When Needed

### Select Print Style

- <1> Standard Pica 80
- <2> Pica HI Qual 80
- <3> Elite HI Qual 96
- <4> Condensed 136
- <5> Italic
- <6> Expand width
- <7> Escape

8000

THE STYLE MENU GROUP, (above is menu), MERGES from the FORMAT MENU for printer Style Changes. When a printer style is changed via menu elective PROMPTS present "a Matching Line Length" and asks for input of "Desired Line Length" for the newly selected style. Then if DOT MATRIX Printer is in use, the DOT MATRIX PRINTER CONTROL GROUP at line #8034 is used to make the printer change its print style. Line #8034 and 8035 CANCELS all printer styles back to the printer's without resetting the TOP OF FORM. Lines 8036-8054 affects the style change at the printer. Line 8080 is a "PRINTER READY" check which loops itself if the printer is off line (busy). You should TURN PRINTER ON, then use its panel switch to place it OFF LINE. Then type: PRINT IN 127 ENTER. EDIT line #8080 and replace "253" with the number given from PRINT IN 127. Different TS-2068's give different results of IN 127.

The PRINT STYLE MENU and PRINTER CONTROL SECTION. The printer Control section, lines 8034-8110 is for EPSON COMPATABLE Dot Matrix Printers. Other type of printers will require EDIT and Adjustments of the control codes to cause the same effect with the printer. Consult Printer Manual for its CONTROL CODE COMMANDS.

Lines 8034 Through 8110  
are for DOT MATRIX Printer  
Auto Style Changes.

IF you have printer problems,  
Just Elect "Daisy Wheel"  
During FIRST ON PROMPTS.  
This will set PR=2 and let you  
do printing with MANUAL set  
print styles. (While you ARE  
struggling with editing these  
lines for your printer).

## Summary of Daisy.B6 Revisions (Nov. 90)

1. The Daisy Sequential Files functions were deleted. These were replaced with provisions for Loading and using the new Universal Data Base Manager (Udbm.B6) software which has expansive Sequential Files management, as well as many other Data base and word processing capabilities. The Daisy and Udbm.B6 programs are now fully integrated.
2. Inclusion of the program "Uptr.B6", an all-purpose Word Processor that has unique capability to print from 1 to 50 "mixed Disk Files" in sequence.
3. The "Mail.B6" program has been replaced with "Dbx.B6", which has expanded Sorting and Search functions, plus "user defined" Data base fields. Dbx.B6 is a "Universal" Records management Software.
4. "Turbo 3" Mode added for general purpose INPUT, Editing, and Printing speed.
5. Expansion of the "Cdbse.B6" program to include cross transfer of data files, Code to Chr Array and Chr Array to Code files.
6. Inclusion of Disk Library management during program operation and the option to load the Toronto Disk Management Utilities (If on hand).
7. Expanded "Macro.B6" program for Screen Files management.
8. Minor changes in the Data Input functions and changes in some Menus.
9. Addition of Automatic Headers during Data Input and corresponding inclusion in the Word processors.
10. The Format and Print Style Menus have been modified to improve user interface.

I believe that the above revisions makes the Daisy software much improved and will be welcomed by those who are already using the program.

The coordinated ensembles "Daisy.B6" and "Udbm.B6" provide an array of Data Base management and Word Processing functions that "cross use" each program's data bases.

I hope that you enjoy using these programs. Feel free to call if you have questions.

Bill Jones (904 871 4513)

# The Wild Ensemble

## Daisy.B6 and Udbm.B6 Softwares

Both of these ensembles are quite large. Both include a group of auxillary programs to allow the user to manage a disk library while the software remains "On-Board". All together there are more than a dozen individual programs that can be used for specific purposes. Some of these are "software length". Others are Utilities. All are managed by the Menus of the main programs. Some LOAD and others are Merged. The Merged programs are Deleted when their jobs are finished. The LOADED programs offer a route back to the main software.

A large variety of Data Bases can be created and managed. A "Main" Word Processor is included in both the Daisy and Udbm program, plus several other word processors are included in each of the two main programs. There is no type of Data File that the TS-2068 CAN HANDLE that these programs cannot manage and print-out. The capability includes the management of all data files created by all TS-2068 softwares.

A Major ANNEX to the Daisy.B6 software is the "Dbx.B6" RECORD File Program. Dbx.B6 creates and manages a 3 dimensional Data Base for RECORDS. Some typical uses are: Mail Files, Inventory Files, Telephone Listings. The Daisy program uses this data base for Automatic Form Letter addressing and can print an endless stream of individually addressed letters by pulling in any number of the mail files from disk. All of a Record File, or selected records, may also be included within a text print out.

Another Daisy Data base is the "Outline File". This is TWO Dimensioned Data bases that are created as "major topics and minor topics of each major", just as you would see in a College Outline. The Outline data base is word processed alone or can be inserted between paragraphs of any other data base being word processed. Dbx.B6 is a sophisticated program having full Edit as well as Search. The dimensions of the data base can be set by the user, which makes it usable for just about any type of record file.

Why TWO major Data base programs? Well, Daisy includes a wider variety of FORMATTING capability for Text Data Printing. Daisy has been tried and proven by three years of publishing Update Magazine and is ideal for such work. Udbm.B6 was designed to be a more "all inclusive" Data Base Manager. I kept getting text articles prepared in Mscript or Tasword (Code Files). Now both are good softwares but lack the formatting capabilities needed for publishing. So Udbm was started to be a "convertor" for those code files. But you will see that the program grew and grew like Pinnocio's nose.

## Daisy and Udbm.B6 Discussed Separately

DAISY.B6: This program is designed for the writer who needs to have a variety of formatting choices. Many automatic functions are provided to allow the writer to concentrate upon creating the text, not worrying about how it will be printed. INPUT TYPING presents the text "un-formatted" on the 32 column screen. The TS-2068's screen presents large characters that are easier to read than most computers small characters. When programming is used to reduce the character size we run into "squint" problems. Most TS-2068 software does this to try to give a Formatting capability on screen. The very widest column line practical with the TS-2068 is 64 character lines and even that character size is difficult to read on screen.

PRINTERS easily can print up to 150 or more characters per line. 64 col screens must "wrap the print" to show a formatted line. These problems, and more, made it sensible to use the TS-2068 screen "as Timex Designed it to be used", an easy to read 32 column line. So, the text that is presented on screen during typing is "Just for Reading, Editing, and to keep the typist informed about what has been written. The Text presentation during typing has no relation at all as to how it will be print Formatted. We do not insert codes for formatting, nor do we pad spaces for effect (indenting). Just type away and create good text.

PRINT FORMATTING is done by MENU electives. A "Print Style" menu is punched up from the Main Daisy Menu. When a Style of print is elected PROMPTS ask for LINE LENGTH. Then the program computes the left margin for the text to be CENTER PRINTED on the page. Then we are ready to select a Print-Out. Thats all there is to it! BUT, another menu is provided, "The Format Menu", where special formats can be elected. Such formats as Column Print, Off-set printing, Block Indent of Paragraphs, Double Line Spacing, can all be elected.

## THE FORMAT MENU and THE STYLE MENU

PRINT FORMATTING is done by MENU ELECTIVE. The normal Default settings (when Daisy first Loaded) is for Pica Style (80 Chr max Line), 60 chrs per line, and Tab 10, to produce a CENTERED TEXT PAGE. The printer when first turned on is set to Draft Quality and Pica 10 Cpi. When the "Style Menu" is punched up and a new Style is set, the software sets the new style, PLUS "Letter Quality". Other automatic settings occur to re-set margins and page center for the new style to result in CENTERED text printing. Options are given to re-set the line length. Also, the Style menu gives "Matched Line", which means the line length for the new style selected that will "Match" the margins of the print style previously used.

So, the STYLE MENU is an extension of the FORMAT menu, to be used to quickly change a print style. The Format menu is used to change the page format for a Style of print that has been elected. PRINTER'S Default settings are for 11/64 inch LINE SPACING which produces a page of 60 lines (max). Most printers can be manually changed by either the front panel switches or by changing the settings of a couple of Dip Switches. When the Format menu is used there is an option to allow such

manual printer change to produce wider or narrower line spacing for different sizes of print. For example, we can change the printer to 1/8 inch spacing to produce about 80 lines per page. That looks pretty good when using condensed print style.

Or, we should change printer to 1/4" spacing to use for EXPANDED Height printing. That would give pages of 48 Lines (max). After making such line spacing changes at the printer the Format menu asks for "Lines Per Page", and gives a recommended number. This coordinates the printer settings with your desire for line formatting. OTHER choices at the Format menu provide for OFF SET Printing, Column Printing, Double Line Spacing, and Block Indent. Block Indent selection lets us print a paragraph that is indented five spaces IN OR OUT in reference to previous margins.

The election of COLUMN Print results in the printer being automatically set to MICRON style (120 Chr Max Lines), and Column width of 50 characters per line, and TAB 8. When printing is started the page is printed in alternate left and right columns for paste-up. Options allow for starting the print at any place down on the page (for other paste-ups). This feature has been a valuable one for publishing Update Magazine. OFF-SET printing options (using any print style) are available. While the software defaults to CENTER PRINT, prompts allow you to input any margin.

## The DAISY DATA BASE FILES

TEXT FILES: The Main Data Base File consists of two Dimensioned Character array, the H\$ file and the I\$ file. These files are dimensioned (7,XXXX), with "XXXX" being your choice of paragraph size. Generally the FREE MEMORY status allows about 15K of data files to be created and edited. If we are only needing to create TEXT this allows for about 14 lengthy paragraphs to be created "IN MEMORY" (Approximately 1000 chrs per paragraph). But MEMORY is used mostly as a BUFFER, with the disk drive used for quick storage.

The Daisy scheme is to use disk drive "as if it were an extension of the TS-2068's memory to pull in data as needed for printing or editing, or to WORD PROCESS the data files in disk, either SINGLY or IN CHAIN. So, FREE MEMORY is of little concern. The TWO character arrays are given so that one can manage several other data base files in memory, concurrently with the main TEXT File. Where FREE memory is of concern is when we plan to use the other two data bases concurrently (Outlines or Record Files). Then we would create and manage "just one" of the two text files.

Continuous text files then would be saved in single files of 7 paragraphs with the file numbers "in sequence". The exclusive file name for the text files is "#.A\$". Example: "1.A\$", "2.A\$", "3.A\$", etc. Suppose, for talking purposes, we have elected 900 as the "paragraph length" and have typed 9 paragraphs to memory. The first 7 would be stored in memory in the Memory file "H\$(7,900)". Paragraphs #8 and #9 would be stored in the memory file "I\$(7,900)", and there would be 5 VACANCIES in the I\$(7,900) memory file. The Text of the paragraphs is of random length.



The WORD PROCESSOR "does not care" what length the paragraphs are, nor if ALL CELLS of the character array files are filled with data. When the files are Word Processed the software SENSES "Data END" and proceeds to word process the next Paragraph Cell that does contain data. This "sensing" looks for more than TWO SPACES within the paragraph text. So, it is important that SPACES NOT BE PADDED within the Text for effect such as INDENTING or internal Formatting. Formatting is automatically done by the software and to insert padded spaces will result in truncation.

## HEADERS

Both Daisy and Udbm.B6 ensembles provide a common way of using pre-planned Headers. When we want to include a header we simply, during Input Typing, Use the British Pound as the first character of a Line. This must be a an isolated line and not one within a passage of text, and the line must be saved Alone. With the Daisy program this means entering a single line as we do a larger paragraph. In other words, the Header line would be Saved as a Paragraph of text. Later, during Word Processing, the Pound symbol will be sensed and the line will be CENTER Printed as a Header, and in Double High Characters. Then the Word processor will proceed to print the following paragraphs of the same data base.

## The Dbx RECORD Data Base

This data base was mentioned before. Now we will discuss it at more length. Originally the Daisy program included a program annex called "Mail Merge", and the manual describes that program. "Dbx.B6" is Mail Merge EXPANDED to include SORTING and SEARCH. Also, the data base was made more flexible in that the DB can be "user dimensioned". Dbx.B6 is a full fledged software that operates alone to create and word process RECORDS. A typical record is a Mail File or an Inventory File, or Daily Diary. I suppose that the primary use will be a Mail File consisting of 4 to 7 lines of about 32 characters. Thats the type of file that works well with the Daisy Program for Mail Merge processing. Daisy uses such files to address form letters. Daisy pulls such files from disk (any number of them) and prints letters to ALL names and addresses.

I will use the example of how I have used the Dbx.B6 Record File. Update Magazine had a name listing of over 500 subscribers. Each magazine issue needed labels printed. The Mail File was in 7 Disk Files named "m1.A\$" through "m5.A\$", 100 (or fewer) names and addresses per file. The Dbx.B6 program was loaded and the labels printed for the magazine mail out. If I got a letter from one of the subscribers, I'd punch up the files and use the SEARCH mode to find the subscriber's address, and what type of equipment (disk system) he used. On a few occasions I have mailed out form letters to all subscribers. A single letter was prepared. Then the Daisy program was used for addressing the letters and envelopes. This mode of the Daisy program is menu elected.

Then Prompts ask for "Starting Mail File #, and ENDING Mail File

#". Punching in "1 for Start and 7 for End" resulted in 500 "individually addressed" form letters being printed "Non-Stop". The administration of Name Lists is greatly assisted by the Dbx.B6 program. The "User Defined" Dimensions of the data base makes it very useful for just any set of records needed. then there is a very fine SORTING capability. The Data Base can be Sorted "By any Line number of the Records", and the sort can be by either "First Group or Last Group" in the sorting line. Example: The mail file can be sorted by LAST NAME of LINE #1. Or, it can be sorted by "Last Group of Line #4", which is the Zip Code. An Inventory file could be sorted by Stock Number or Noun. Dbx.B6 is a very useful program and I hope that you will appreciate it.

THE OUTLINE DATA BASE: One of the Update Magazine subscribers is a Biology Professor who needed an Outline program to produce school Biology outlines. I thought it was a good idea and the Outline section of the Daisy program is the result. This is a "twin data base". The major topics are held in a two dimensioned character array "K\$(xx,yy)". Example: a group of 40 major topics with a line length of 48 characters each. The dimension would be: K\$(40,48). Then the Minor Topic Data base must conform to "40" major topics. If we want 4 sub-divisions of each Major topic, then the 3 Dimensions would be "V\$(40,4,43)", because the Minor topics will be indented five spaces. So there we have a typical Twin Outline data base. The dimensions of both data bases are set by Prompts. You can set up the dimensions needed for your own purpose.

## Auxillary Basic Programs

Two of these larger programs have already been discussed (The Dbx.B6 "Records Data Base Manager", and the "Uptr.B6" Universal Word Processor). Next, we will discuss 6 more Basic Programs which are Merged to the Daisy and Udbm programs, BUT are usable as separately Loaded programs.

Macro.B6: This neat program creates and manages Disk Screen Files. The screen is used as a Data Base to be created, edited, and SAVED to disk. Disk "Screen Macros" are re-loaded by this program, edited if needed, re-colored or de-colored if needed, or Printed to paper, or re-Saved. Actually ANY screen file in disk may be managed in the same manner. Both Daisy and Udbm.B6 programs do this screen file management by MERGING the "macro.B6" program. After finishing with Screen management the program is deleted and the main menu of the Daisy or Udbm program is returned. Such file management is done without disturbing the main program's data bases in memory. The "Macro.B6" program can also be Loaded and operated alone. "Paint.B6", which is a part of the Macro.B6 program was pulled out and Saved to be used alone. This little program lets you pull in Disk Screen files and "Paint" them.

"Cdbse.B6": This is a program that completely manages Code Ascii Files from disk. Three common types of code ascii files managed are: Mterm Files, Tasword Files, and Mscript Files. Such code files are loaded from disk, their byte length and formatted line length found. Then the files may be printed to paper. OR, the files may be "Pagenated", that is, "Transposed from Code File Data to Character Array Files of Lines." An annex of the Cdbse program is "xfr.B6" which is used

to "Codenate" character array files as used by Daisy and Udbm. Several types of Character array files are transposed into either Mscript or Tasword Code Ascii Files. The genesis of the Cdbse.B6 program was my need to process Code Ascii Files sent as text to be published in Update Magazine. It is an interesting program but likely wont be used much except by the consumates.

"Dbasel.B6" is an experimental program, still under development. Although not finished it has enough operating functions to be useful for fast sorting of a multi dimensioned Record File. Use it at your peril. Generally, Dbasel.B6, when finished, will take, say 20 Disk Record Files of three dimensions, and sort them rapidly "against each other". The complexity of programming is enoromous, and is amplified by the requirement to mix Basic Language Disk Management with Machine Code Sorting. Right now, if everything is set up right, the program will sort a 150 Mail List Record file < O\$(150,7,31) > in about 5 seconds. Now to make it do 20 such record files in about 5-10 minutes!

"Varsv.B6": This program SAVES and LOADS the Whole TS-2068 variable file. One of the Data Base schemes of the Daisy program is to use the variable file for "Multiple Data Base generation". Example: A text file in the H\$() Chr Array, plus, An Outline file in the K\$() and V\$() Chr Arrays, and a Mail File in the O\$() character array. "Varsv.B6" SAVES the whole Smear and Re-Loads any vars file saved with this program. To simplify operations, a single Menu key-punch does the SAVE or LOAD. All you need to apply is the file name. This vars save load utility was written up in an article in the 1889 issues of Update Magazine.

FOUR SMALL PROGRAMS: "Init.B6" is a program that initializes the LKDOS system for one of four Centronics Interfaces, Prompts for PRINTER STYLE, and Sets the style, plus Letter Quality mode. \*\*\*\* "Cat.B6" is a program that PRINTS OUT repeat or single Disk Catalogs- to Tab position on a page. \*\*\* "See.B6" is a tiny program that "looks at" disk Screen Files. \*\*\*\* "Labels.B6" is a small but very useful program to load and print a series of mailing labels. \*\*\* So, there you have a repertorie of useful utilities to use at your pleasure. Most are available by Menu elective at one or the Daisy or Udbm.B6 program menus, and all can be Loaded to use separately. User Clubs are granted permission to publish or use these utilities for any purpose desired. Bill Jones.

## Concepts

Whereas, The TS-2068 has a limit of 38652 bytes of Free Memory, and Whereas, to get anything done with a computer one needs more Free Memory than that, and Whereas one Larry Kenny (Larken Electronics) designed a DOS that allows us to "Overlay" programs with the MERGE function, and Whereas, Larry also included a good SCREEN FILE Load and Save, which saves about 700 bytes of Free Memory with each usage, and Whereas, the use of both of these functions allow "UNLIMITED" software length- even larger than the WHOLE TS-2068's RAM CAPACITY, NOW COMES a Software ensemble that requires FOUR 360K Diskettes in order to contain the ensemble of programs.

## FIVE Coordinated Softwares

The five softwares are: "Daisy.B6", "Udbm.B6" (Universal Data Base Manager), "Dbx.B6" (Record Data Base Manager), "Uptr.B6" (Universal Word Processor), and "Cdbse.B6" (Code Ascii File Manager). Each of these programs have at least one Word Processor. Daisy and Udbm.B6 have three WP's each. The concept of a single word processor to print various data files is archaic because each type of data base needs special printing formats.

DO YOU HAVE to have data "in a program" in order to word process? Ans: No! Just Load up one of the five programs and GET DATA FROM DISK FILES. Of course someone must create the files. We are leading into the subject of "DISK DATA BASE MANAGEMENT". All of the "Wild Ensemble" of programs are designed with Disk Management as the foremost feature. The main Daisy Data Base is two RANDOM Length data files. We can create 14 large paragraphs consecutively without worrying about a SAVE. Then BOTH files are SAVED with one key punch. AND, the file names are "Consecutive Numbers". The two files are then cleared from memory and we keep on writing. Such "Create, SAVE, Clear, CREATE" sequence allows us to do a long manuscript with hardly an interruption in thought.

BUT, sometimes a data base of PARAGRAPHS wont fit our need. The Daisy program has a "RECORD" Data base and a "OUTLINE" Data Base. We can create these data bases via menu elective which shifts us into another INPUT Mode. OR- we can elect to use the Udbm.B6 Input Modes. A

transition menu allows us to select "The Universal Data Base Manager". Then Udbm.B6 loads in. The Udbm.B6 program has no less than SIX different Data Base Files to create and manage. Each has Word Processing as well. HALF of these data bases are "SEQUENTIAL FILES", using the LKDOS Sequential Files mode.

## WHAT ARE Sequential And RANDOM FILES?

Every Input to the Computer is a RANDOM Input. We can control the length of the input by simply stopping. A Software program can impose limits of the length of input. In the case of both Daisy and Udbm.B6, the USER, that's YOU, sets the length limit of the Input, which makes RANDOM Data Files a Finite Maximum Length. If we are creating PARAGRAPHS, we set a Length Limit that is comfortable. Then we stay within that limit. But the text is of Random Length Paragraphs. If we set a LINE LENGTH Limit for Line Files then the software helps us to stay within that limit. Still, the Lines are of Random Length (hopefully not too ragged). The Word Processor will even them out by justification.

SEQUENTIAL FILES are Random length Typing bursts that are sent to an Opened Disk File. We use the Udbm.B6 program to create Sequential Files. There are three types of Sequential Files created. 1. Random Length Paragraphs. 2. Line Files of fixed Line length and Random in total number. 3. Lines of Fixed length and "Page Length in number". These files are easily created and can be from small to enormous in total length. They are "read back" from disk for word processing, and can be "chained", ie, several disk files word processed in a single print-out.

WORD PROCESSING: Both Daisy and Udbm.B6 use "each other's" Data files from disk. Daisy has a "User Programming Section" that allows all types of files to be "Sequenced". Disk Data Files are read into the word processor and printed in unlimited "mixtures" of file types. The "usrpgm" section of Daisy is very versatile, allowing multiplicity of formatting, large or small headers, and the scheduling of just any data print out that the TS-2068 is capable of processing. But- one must learn to use the "usrpgm" section, and it may be a bit more than non-programmers can handle.

## Enter the "Uptr.B6" Program!

"Uptr.B6" is abbreviated for "The Universal

Word Processor". This program is available from both Daisy and the Udbm.B6 program. When in Daisy and Printing is Elected a choice is to use the Uptr.B6 Word Processor. This program asks for INPUT of a Disk File name- or two- or 10- or 50 file names. We input the file names and then the program begins printing all files, a single short letter, or a 20 page manual. The file types can be mixed, as: A Daisy Paragraph file or several, A Sequential Line File or several, A Sequential Paragraph File or several, etc. The Disk File, or Files, are Center Printed to our desired line length. Headers are included if present in the disk file.

## Screen and Page Macros

"Macro.B6" is a program that is used by both Daisy and Udbm to create "Screen Macros", a data base of 22 lines and 32 characters per line. You can even create a nice MENU for your programming. Actually a screen holds a fairly large amount of text, and can be used for text generation. "Macro.B6" contains its own little word processor that allows you to "place the data" anywhere on a page. Macro.B6 also includes EDITING of Disk Screen Files and "Painting them with color". If we are using the Macro.B6 program from the Daisy Menu, its use does not in any way disturb the existing data bases in the Daisy program. --- "PAGE MACROS" is another data base given in the Udbm.B6 program. Here we "design a page", create it, save it to disk as a page, recall it from disk for print-out as a page. Its Flexible Line length lets us prepare pages to suit any print style.

## Different Concepts, Daisy and Udbm.B6

Daisy is a program designed for all purpose Text Data Base Management, with unlimited choices of word processing formats. Many helpful Automatic features allow one to quickly print out a document or a letter. Daisy manages data well in both Memory and Disk Drive and has a myriad of automatic Disk Coordinating functions. The "Dbx.B6" program provides great support for form letter printing and label printing. THE UDBM.B6 program is a "universal data base manager" that can be used to create many different types of data bases. Its main word processor is used to print out its own data bases AND those created with Daisy, but is not quite as sophisticated in formatting choices. BOTH programs "alternately" uses the "Uptr.B6" (Universal Word Processor) for "Chain Printing" mixed Disk Data Files. Generally, the Udbm.B6 program is more "Disk

Interactive", but both programs manage disk drive completely.

## Coordination with other Software

The program menus provide paths to use both Supplied Utilities and others that may be in our library. A full set of Disk Management Utilities are available from the Daisy Menus. We can escape what we are doing and elect "Disk Management". The Functions are available without disturbing any existing data in memory, or any pre-established format set-up. These disk utilities include: 1. Viewing Catalogs, or Copying them. 2. Viewing Disk Screen Files, Erasing Disk Files. 3. Creating Screen Files (macros), printing them, editing them, coloring them, erasing, Saving, and Cross Saving. 4. View or Print Disk Data Files. A selection allows us use the LKDOS capability to directly view or LPRINT disk data files (un-formatted). This is very valuable to use disk files for reference during our toils of writing a document. When Finished, we return to Daisy "Status Quo Ante" and resume what we were doing, but better informed.

## The Learning Curve

Well heck! What can I say? The Ensemble of Daisy, Udbm.B6, and the other supporting programs really makes up a "Software System". Each program is self documented with help files. But there are literally scores of functions to get familiar with. Once we do a function successfully we can do it again without study. The 30 or so menus are each "subjective" which helps greatly. Best to start out SIMPLE and do some letters first. Then tie in the Mail Merge concept and send a Christmas Letter to all of your family and friends (I hate them "me me" letters, dont you?). Suffice it to say that this ensemble of programs will do just any Administrative Job that you need to do. There are enough functions to keep you on the learning curve throughout a year or so. And each time you use the programs you'll learn something new. I'm available for help calls and I enjoy talking to folks. Tel: 904 871 4513. Enjoy! Bill Jones.



Disk Name : Daisy.B6

|           |     |           |     |
|-----------|-----|-----------|-----|
| Daisy.B6  | 004 | init.B6   | 001 |
| repp.B6   | 001 | varsv.B6  | 001 |
| stymm.B6  | 001 | otln.B6   | 001 |
| fomat.B6  | 001 | wdpro.B6  | 001 |
| tb.B6     | 001 | inpt.B6   | 001 |
| see.B6    | 001 | cat.B6    | 001 |
| usrpgm.B6 | 001 | pt.B6     | 001 |
| paint.B6  | 001 | st.B6     | 001 |
| dc.B6     | 001 | fm.C1     | 002 |
| mg1.C1    | 002 | cpr.C1    | 002 |
| tb.C1     | 002 | 5.A\$     | 002 |
| otl.C1    | 002 | ed.C1     | 002 |
| rp.C1     | 002 | sts.C1    | 002 |
| ot.C1     | 002 | pg.C1     | 002 |
| sm.C1     | 002 | ps.C1     | 002 |
| fo5.C1    | 002 | if.C1     | 002 |
| lm.C1     | 002 | prr.C1    | 002 |
| dc.C1     | 002 | ab.C1     | 002 |
| xx.B6     | 004 | olt.C1    | 002 |
| px.B6     | 001 | 2.A\$     | 001 |
| var.B6    | 001 | srch.B6   | 001 |
| cat.C1    | 001 | 01831.C2  | 001 |
| 6.A\$     | 002 | edt.B6    | 001 |
| 7.A\$     | 002 | AUTOSTART | 001 |
| macro.B6  | 001 | cinfo.C1  | 002 |
| vcT?.CT   | 002 | word.B6   | 001 |
| m1.A\$    | 001 | macro1.C4 | 002 |
| macro2.C4 | 002 | Init.B6   | 001 |
| Dbx.B6    | 001 | 1.A\$     | 002 |
| varset.B6 | 001 | db1.C1    | 002 |
| mg1.C1    | 002 | exit.C1   | 002 |
| m16.C1    | 002 | m2.C1     | 002 |
| v.C1      | 002 | vgh2.CV   | 005 |
| taswi.Cx  | 001 | cclear.C1 | 007 |
| dp2.C1    | 002 | DP.C1     | 001 |
| sort.B6   | 001 | 01.A\$    | 001 |
| 02.A\$    | 001 | dbase1.B6 | 003 |
| rpt.C4    | 002 | cd6.C4    | 002 |
| xx.C4     | 002 | cd10.C4   | 002 |
| add1.A\$  | 001 | add2.A\$  | 001 |
| v2.CP     | 001 | 3.A\$     | 002 |
| Cdbse.B6  | 001 | cd2.C1    | 002 |
| xfr.B6    | 001 | cv.C1     | 002 |
| pgn.B6    | 001 | cd7.C4    | 002 |
| Uptr.B6   | 002 | vcC?.CT   | 002 |
| 4.A\$     | 000 |           |     |

LARKEN LKDOS 1986  
Track/Side 080/002  
Total Files 091  
Free Blocks 007

Disk Name : "Udbm.B6"

# Universal Data Base Manager

By: Bill Jones  
1317 Stratford Ave  
Panama City, FL  
32404  
Tel: 904 871 4513

HELP Screens Selection at Menus  
AUTO LOAD or LOAD "Udbm.B6"

|           |     |           |     |
|-----------|-----|-----------|-----|
| AUTOSTART | 001 | sr.C4     | 002 |
| Udbm.B6   | 005 | Cdbse.B6  | 001 |
| macro.B6  | 001 | pgn.B6    | 001 |
| Init.B6   | 001 | xfr.B6    | 001 |
| ptr.C4    | 002 | paint.B2  | 001 |
| com.C4    | 002 | cat.C1    | 001 |
| cclear.C1 | 006 | DP.C1     | 001 |
| see.B2    | 001 | Sql8.A\$  | 001 |
| db1.C1    | 002 | db4.C1    | 002 |
| sq.C1     | 002 | pp.C1     | 002 |
| Labels.B6 | 001 | cv.C1     | 002 |
| cd2.C1    | 002 | cd2.C4    | 002 |
| cd6.C4    | 002 | macro.C4  | 002 |
| cd8.C4    | 002 | cd3.C4    | 002 |
| cd9.C4    | 002 | cd4.C4    | 002 |
| cd5.C4    | 002 | cd7.C4    | 002 |
| xx.C4     | 002 | cd10.C4   | 002 |
| 15.C4     | 002 | 10.C4     | 002 |
| 11.C4     | 002 | Sql9.A\$  | 002 |
| Sql2.A\$  | 001 | first.A\$ | 002 |
| Sqp1.A\$  | 002 | Pg1.A\$   | 001 |
| Pg2.A\$   | 001 | Pg4.A\$   | 001 |
| Pg3.A\$   | 001 | Pg5.A\$   | 002 |
| Pg33.A\$  | 001 | Pg10.A\$  | 001 |
| Pg34.A\$  | 001 | Pg15.A\$  | 001 |
| Pg16.A\$  | 001 | Pg17.A\$  | 001 |
| Pg20.A\$  | 001 | Pg21.A\$  | 001 |
| Pg22.A\$  | 001 | Pg6.A\$   | 001 |
| Pg50.A\$  | 001 | Pg24.A\$  | 002 |
| Pg60.A\$  | 001 | Pg61.A\$  | 001 |
| 1.A\$     | 001 | 2.A\$     | 001 |
| vcC?.CT   | 002 | Tas11.CT  | 002 |
| Tasp50.CT | 001 | Tas151.CT | 001 |
| Tspg53.CT | 001 | Msl1.CM   | 002 |
| Msl40.CM  | 001 | Msp60.CM  | 001 |
| Mpg51.CM  | 002 | npt.C4    | 002 |
| gotos.C4  | 002 | 3.A\$     | 001 |
| Sql3.A\$  | 001 | Pg70.A\$  | 001 |
| Sqp2.A\$  | 002 | Sqlp4.A\$ | 001 |
| Sql5.A\$  | 002 | Pg72.A\$  | 001 |
| 95.A\$    | 001 | Pg80.A\$  | 001 |
| Sqp3.A\$  | 001 | Sql6.A\$  | 001 |
| Sql7.A\$  | 001 | xx.C1     | 002 |
| Uptr.B6   | 002 | rpt.C4    | 002 |
| Sqp5.A\$  | 003 | Sql10.A\$ | 001 |